

dsPIC30F2010
数据手册

高性能 16 位
数字信号控制器

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下， Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是 “牢不可破”的。

代码保护功能处于持续发展中。 Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC® MCU 与 dsPIC® DSC、KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外， Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。



dsPIC30F2010

28 引脚 dsPIC30F2010 增强型 16 位闪存数字信号控制器

注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

高性能改进型 RISC CPU：

- 改进的哈佛架构
- 优化的 C 编译器指令集
- 83 条具备灵活寻址模式的基本指令
- 24 位宽指令， 16 位宽数据总线
- 12 KB 片内闪存程序空间
- 512 字节片内数据 RAM
- 1 KB 非易失性数据 EEPROM
- 16 x 16 位工作寄存器阵列
- 最大 30 MIPS 工作：
  - DC 至 40 MHz 外部时钟输入
  - 4 MHz-10 MHz 振荡器输入，带 PLL （4 倍频、8 倍频和 16 倍频）
- 27 个中断源
- 三个外部中断源
- 每一个中断具有 8 个用户可选择的中断优先级
- 4 个处理器异常和软件陷阱

DSP 引擎特性：

- 模寻址和位反转寻址模式
- 两个 40 位宽的累加器，具备可选饱和逻辑
- 17 位 x 17 位单周期硬件小数 / 整数乘法器
- 单周期乘 - 累加 （MAC）操作
- 40 级桶形移位寄存器
- 双数据取

外设特性：

- 高灌电流 / 拉电流 I/O 引脚：25 mA/25 mA
- 3个16位定时器/计数器；可选择将16位定时器配对组成 32 位定时器模块
- 4 个 16 位捕捉输入功能引脚
- 2 个 16 位比较 /PWM 输出功能引脚
  - 可提供双比较模式
- 3 线 SPI 模块 （支持 4 种帧模式）
- I^2C™ 模块支持多主器件/从模式，支持7位/10位寻址
- 带 FIFO 缓冲区的可寻址 UART 模块

电机控制 PWM 模块特性：

- 6 个 PWM 输出通道
  - 互补或独立的输出模式
  - 边沿和中心对齐模式
- 4 个占空比发生器
- 具备 4 种工作模式的专用时基
- 输出极性可由编程设定
- 用于互补模式的死区时间控制
- 手动输出控制
- 用于同步 A/D 转换的触发器

正交编码器接口模块特性：

- A 相、 B 相和索引脉冲输入
- 16 位向上 / 向下位置计数器
- 计数方向状态
- 位置测量 （x2 和 x4）模式
- 输入端具有可编程数字噪声滤波器
- 备用 16 位定时器 / 计数器模式
- 位置计数器计满返回 / 下溢中断

模拟特性：

- 10 位模数转换器 （ADC），具备以下特性：
  - 1 Msps 转换速率 （对于 10 位 A/D）
  - 6 个输入通道
  - 休眠和空闲模式下仍可进行转换
- 可编程欠压复位

特殊数字信号控制器特性：

- 增强型闪存程序存储器：
  - 对于工业级温度范围，最少擦写次数 1 万次，典型擦写次数 10 万次。
- 数据 EEPROM 存储器：
  - 对于工业级温度范围，最少擦写次数 10 万次，典型擦写次数 100 万次。
- 软件控制下，可自行再编程
- 上电复位 （Power-on Reset， POR）、上电延时定时器 （Power-up Timer， PWRT）以及振荡器起振定时器 （Oscillator Start-up Timer， OST）
- 零活的看门狗定时器 （Watchdog Timer，WDT），片内带有低功耗 RC 振荡器，以便可靠地工作
- 故障保护时钟监控器：
  - 检测时钟故障，可切换至片上低功耗 RC 振荡器
- 可编程代码保护
- 在线串行编程 （In-Circuit Serial Programming™， ICSP™）功能
- 可选择的功耗管理模式：
- 休眠、空闲和备用时钟模式

CMOS 技术：

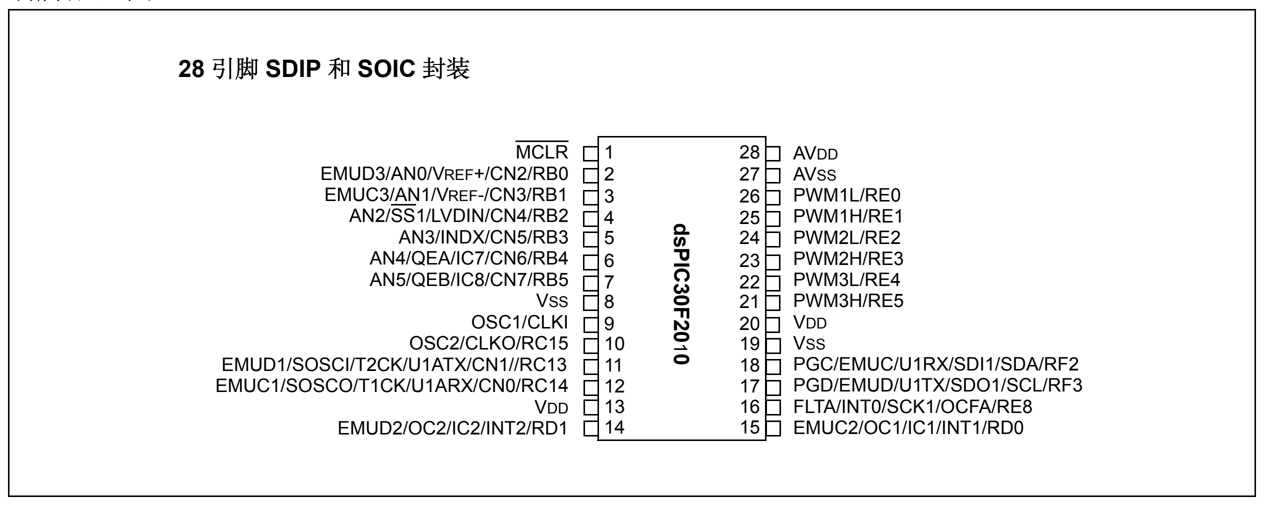
- 低功耗高速闪存技术
- 宽工作电压范围 （2.5V 至 5.5V）
- 工业级和扩展级温度范围
- 低功耗

dsPIC30F 电机控制和电源转换系列器件 *

器件	引脚数	程序存储器字节数 / 指令数	SRAM 字节数	EEPROM 字节数	16 位定时器	输入捕捉	输出比较 / 标准 PWM	电机控制 PWM	A/D 10 位 1 Msps	正交编码器模块	UART	SPI	I^2C^{TM}	CAN
dsPIC30F2010	28	12K/4K	512	1024	3	4	2	6 通道	6 通道	有	1	1	1	–
dsPIC30F3010	28	24K/8K	1024	1024	5	4	2	6 通道	6 通道	有	1	1	1	–
dsPIC30F4012	28	48K/16K	2048	1024	5	4	2	6 通道	6 通道	有	1	1	1	1
dsPIC30F3011	40/44	24K/8K	1024	1024	5	4	4	6 通道	9 通道	有	2	1	1	–
dsPIC30F4011	40/44	48K/16K	2048	1024	5	4	4	6 通道	9 通道	有	2	1	1	1
dsPIC30F5015	64	66K/22K	2048	1024	5	4	4	8 通道	16 通道	有	1	2	1	1
dsPIC30F6010	80	144K/48K	8192	4096	5	8	8	8 通道	16 通道	有	2	2	1	2
dsPIC30F6010A	80	144K/48K	8192	4096	5	8	8	8 通道	16 通道	有	2	2	1	2

* 本表对 dsPIC30F2010 的外设特性进行了汇总，同时列出 dsPIC30F 电机控制和电源转换系列中其他器件的特性以进行比较。

引脚配置图



28 引脚 QFN 封装

dsPIC30F2010

- 1 AN2/SS1/LVDIN/CN4/RB2
- 2 AN3/INDX/CN5 RB3
- 3 AN4/QEA/IC7/CN6/RB4
- 4 AN5/QEB/IC8/CN7/RB5
- 5 VSS
- 6 OSC1/CLKI
- 7 OSC2/CLKO/RC15
- 8 EMUD1/SOSCI/T2CK/U1ATX/CN1/RC13
- 9 EMUC1/SOSCO/T1CK/U1ARX/CN0/RC14
- 10 VDD
- 11 EMUD2/OC2/IC2/INT2/RD1
- 12 EMUC2/OC1/IC1/INT1/RD0
- 13 FLTA/INT0/SCK1/OCFA/RE8
- 14 PGD/EMUD/U1TX/SDO1/SCL/RF3
- 15 PGC/EMUC/U1RX/SDI1/SDA/RF2
- 16 VSS
- 17 VDD
- 18 PWM3H/RE5
- 19 PWM3L/RE4
- 20 PWM2H/RE3
- 21 PWM2L/RE2
- 22 PWM1H/RE1
- 23 PWM1L/RE0
- 24 AVSS
- 25 AVDD
- 26 MCLR
- 27 EMUD3/AN0/VREF+/CN2/RB0
- 28 EMUC3/AN1/VREF- /CN3/RB1

目录

1.0 器件概述 5
2.0 CPU 架构综述 9
3.0 存储器构成 19
4.0 地址发生器单元 31
5.0 中断 37
6.0 闪存程序存储器 43
7.0 数据 EEPROM 存储器 49
8.0 I/O 端口 53
9.0 Timer1 模块 57
10.0 Timer2/3 模块 61
11.0 输入捕捉模块 67
12.0 输出比较模块 71
13.0 正交编码器接口 （QEI）模块 75
14.0 电机控制 PWM 模块 81
15.0 SPI 模块 91
16.0 I2C 模块 95
17.0 通用异步收发器 （UART）模块 103
18.0 10 位高速模数转换器 （ADC）模块 111
19.0 系统集成 121
20.0 指令集汇总 135
21.0 开发支持 143
22.0 电气特性 147
23.0 封装信息 185
Microchip 网站 197
变更通知客户服务 197
客户支持 197
读者反馈表 198
产品标识体系 199

致 客 户

我们旨在提供最佳文档供客户正确使用 Microchip 产品。为此，我们将不断改进出版物的内容和质量，使之更好地满足您的要求。出版物的质量将随新文档及更新版本的推出而得到提升。

如果您对本出版物有任何问题和建议，请通过电子邮件联系我公司 TRC 经理，电子邮件地址为 **CTRC@microchip.com**，或将本数据手册后附的 **《读者反馈表》**传真到 86-21-5407 5066。我们期待您的反馈。

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

http://www.microchip.com

查看数据手册中任意一页下边角处的文献编号即可确定其版本。文献编号中数字串后的字母是版本号，例如：DS30000A是DS30000 的 A 版本。

勘误表

现有器件可能带有一份勘误表，描述了实际运行与数据手册中记载内容之间存在的细微差异以及建议的变通方法。一旦我们了解到器件 / 文档存在某些差异时，就会发布勘误表。勘误表上将注明其所适用的硅片版本和文件版本。

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

- Microchip 网站 http://www.microchip.com
- 当地 Microchip 销售办事处 （见最后一页）

在联络销售办事处时，请说明您所使用的器件型号、硅片版本和数据手册版本 （包括文献编号）。

客户通知系统

欲及时获知 Microchip 产品的最新信息，请到我公司网站 **www.microchip.com** 上注册。

1.0 器件概述

注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

本文档包含 dsPIC30F2010 器件的特定信息。dsPIC30F 系列器件在高性能 16 位单片机（MCU）架构中融合了大量数字信号处理器（DSP）功能。图 1-1 中给出了 dsPIC30F2010 器件的框图。

图 1-1： dsPIC30F2010 框图

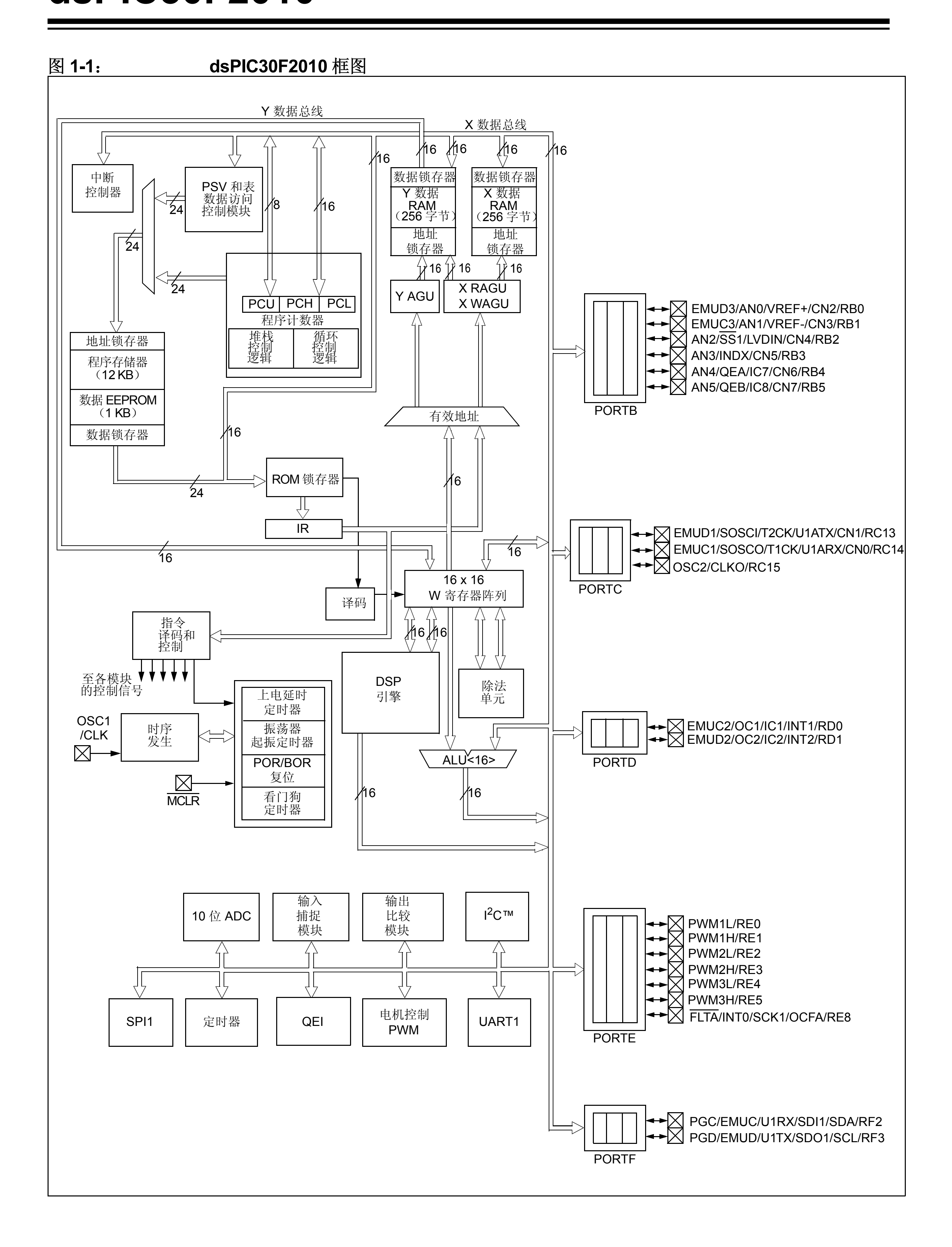


表 1-1 对器件的 I/O 引脚配置和端口引脚的复用功能进行了简要描述。端口引脚可具有多种功能。当发生复用时，外设模块的功能要求可强制替代端口引脚的数据方向。

表 **1-1**： 引脚 **I/O** 描述

引脚名称	引脚类型	缓冲器类型	描述
AN0-AN5	I	Analog	模拟输入通道。
AVDD	P	P	模拟模块正电源。
AVSS	P	P	模拟模块的参考地。
CLKI CLKO	I O	ST/CMOS —	外部时钟源输入。总是与 OSC1 引脚功能相关联。 振荡器晶振输出。在晶振模式下，连接到晶体或谐振器。在 RC 和 EC 模式下，可选择作为 CLKO。总是与 OSC2 引脚功能相关联。
CN0-CN7	I	ST	输入电平变化通知输入。 可通过软件编程设定所有输入为内部弱上拉。
EMUD EMUC EMUD1 EMUC1 EMUD2 EMUC2 EMUD3 EMUC3	I/O I/O I/O I/O I/O I/O I/O I/O	ST ST ST ST ST ST ST ST	ICD 主通信通道数据输入 / 输出引脚。 ICD 主通信通道时钟输入 / 输出引脚。 ICD 第二通信通道数据输入 / 输出引脚。 ICD 第二通信通道时钟输入 / 输出引脚。 ICD 第三通信通道数据输入 / 输出引脚。 ICD 第三通信通道时钟输入 / 输出引脚。 ICD 第四通信通道数据输入 / 输出引脚。 ICD 第四通信通道时钟输入 / 输出引脚。
IC1, IC2, IC7, IC8	I	ST	捕捉输入。 dsPIC30F2010 具有 4 个捕捉输入。输入采用的编号方式有助于保持与引脚数较多器件的输入一致。
INDX QEA QEB	I I I	ST ST ST	正交编码器索引脉冲输入。 在 QEI 模式下为正交编码器 A 相输入。 在定时器模式下为辅助定时器外部时钟 / 门控输入。 在 QEI 模式下为正交编码器 B 相输入。 在定时器模式下为辅助定时器外部时钟 / 门控输入。
INT0 INT1 INT2	I I I	ST ST ST	外部中断 0 外部中断 1 外部中断 2
FLTA PWM1L PWM1H PWM2L PWM2H PWM3L PWM3H	I O O O O O O	ST — — — — — —	PWM 故障 A 输入 PWM1 低端输出 PWM1 高端输出 PWM2 低端输出 PWM2 高端输出 PWM3 低端输出 PWM3 高端输出
$\overline{\text{MCLR}}$	I/P	ST	主复位输入或编程电压输入。本引脚为低电平有效的器件复位输入端。
OCFA OC1-OC2	I O	ST —	比较故障 A 输入 （对于比较通道 1、 2、 3 和 4）。 比较输出。
OSC1 OSC2	I I/O	ST/CMOS —	振荡器晶振输入。当配置为 RC 模式时，为 ST 缓冲器；否则为 CMOS。 振荡器晶振输出。在晶振模式下，连接到晶振或谐振器。在 RC 和 EC 模式下，可选择作为 CLKO。

图注： CMOS =CMOS 兼容输入或输出　Analog = 模拟输入
ST =CMOS 电平的施密特触发器输入　O = 输出
I = 输入　P =电源

表 **1-1**： 引脚 **I/O** 描述 （续）

引脚名称	引脚类型	缓冲器类型	描述
PGD PGC	I/O I	ST ST	在线串行编程数据输入 / 输出引脚。 在线串行编程时钟输入引脚。
RB0-RB5	I/O	ST	PORTB 为双向 I/O 口。
RC13-RC14	I/O	ST	PORTC 为双向 I/O 口。
RD0-RD1	I/O	ST	PORTD 为双向 I/O 口。
RE0-RE5, RE8	I/O	ST	PORTE 为双向 I/O 口。
RF2, RF3	I/O	ST	PORTF 为双向 I/O 口。
SCK1 SDI1 SDO1 SS1	I/O I O I	ST ST — ST	SPI #1 的同步串行时钟输入 / 输出。 SPI #1 数据输入。 SPI #1 数据输出。 SPI #1 同步从动模式。
SCL SDA	I/O I/O	ST ST	用于 I^2C™ 的同步串行时钟输入 / 输出。 用于 I^2C 的同步串行数据输入 / 输出。
SOSCO SOSCI	O I	— ST/CMOS	32 kHz 低功耗晶振输出。 32 kHz 低功耗晶振输入。配置为 RC 模式时，为 ST 缓冲器；否则为 CMOS。
T1CK T2CK	I I	ST ST	Timer1 外部时钟输入。 Timer2 外部时钟输入。
U1RX U1TX U1ARX U1ATX	I O I O	ST — ST —	UART1 接收。 UART1 发送。 UART1 备用接收。 UART1 备用发送。
VDD	P	—	逻辑电路和 I/O 引脚的正电源。
VSS	P	—	逻辑电路和 I/O 引脚的参考地。
VREF+	I	Analog	模拟参考电压 （高电压）输入。
VREF-	I	Analog	模拟参考电压 （低电压）输入。

图注： CMOS =CMOS 兼容输入或输出　　Analog = 模拟输入
ST =CMOS 电平的施密特触发器输入　　O = 输出
I = 输入　　P =电源

2.0 CPU 架构综述

> **注：**本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

本文档对 dsPIC30F2010 的 CPU 和外设功能进行了简要概述。有关这些内容的详细描述，请参见《dsPIC30F 系列参考手册》（DS70046E_CN）。

2.1 内核概述

内核的指令字长为 24 位。程序计数器 （PC）为 23 位宽，其最低有效位 （LSb）始终为 0 （见**第 3.1 节 "程序地址空间"**），最高有效位 （MSb）在正常的程序执行期间被忽略，某些专用指令除外。因此， PC 能够寻址最大为 4M 指令字的用户程序空间。使用指令预取机制来帮助维持吞吐量。使用 DO 和 REPEAT 指令，支持无循环计数管理开销的程序循环结构，而 DO 和 REPEAT 这两条指令在任何时候都可被中断。

工作寄存器阵列由 16 个 16 位寄存器组成，每个寄存器均可作为数据、地址或偏移量寄存器。一个工作寄存器（W15）用作中断和调用时的软件堆栈指针。

数据空间为 64KB （32K 字），被分成两块，称为 X 和 Y 数据存储空间。每个存储块有各自独立的地址发生单元（Address Generation Unit，AGU）。大多数指令只通过 X 存储空间 AGU 进行操作，这样对外界而言数据空间就是单独而统一的。乘法－累加（Multiply-Accumulate，MAC）类双源操作数 DSP 指令，通过 X 和 Y 的 AGU 进行操作，这将数据地址空间分成两个部分（见**第 3.2 节"数据地址空间"**）。X 和 Y 数据空间边界视具体器件而定，用户不能改变它们。每个数据字由 2 个字节构成，大多数指令可以把数据当作字或字节来进行寻址。

对于存储在程序存储空间中的数据，有两种存取方法：

- 数据存储空间的高 32 KB 可以映射到程序空间的下半部分 （用户空间），映射的边界可以是任何 16K 程序字边界，边界由 8 位的程序空间可视性页（ Program Space Visibility Page， PSVPAG）寄存器定义。这使得任何指令都能象存取数据存储空间那样存取程序存储空间，但存取需要一个额外的周期。此外，使用这个方法，只能存取每个指令字的低 16 位。
- 使用任何工作寄存器，通过表读及表写指令，可以对程序空间内的 32K 字页进行线性间接存取。表读和表写指令能够用来存取一个指令字的全部 24 位。

X 和 Y 地址空间都支持无开销循环缓冲区 （模寻址）。模寻址的主要目的是要消除 DSP 算法的循环开销。

X AGU 还支持对目的有效地址进行位反转寻址，对于基 –2 FFT 算法，这极大地简化了输入或输出数据的重新排序。关于模寻址和位反转寻址的详细信息，参见**第 4.0 节 "地址发生器单元"**。

内核支持固有（无操作数）寻址、相对寻址、立即数寻址、存储器直接寻址、寄存器直接寻址、寄存器间接寻址、寄存器偏移量寻址以及立即数偏移量寻址模式。指令与预定义的寻址模式相关联，这取决于其功能需求。

对于大多数指令，内核能够在每个指令周期内执行数据（或程序数据）存储器读取、工作寄存器（数据）读取、数据存储器写入以及程序（指令）存储器读取操作。因此，支持 3 操作数指令，允许在单个周期内执行 C = A + B 这样的操作。

内核包含 DSP 引擎，这极大地提高了内核的运算能力和吞吐能力。它具有一个高速 17 位× 17 位乘法器、一个 40 位 ALU、两个 40 位饱和累加器和一个 40 位双向桶形移位寄存器。累加器或任何工作寄存器中的数据，可以在单个周期内右移 16 位或左移 16 位。 DSP 指令可以无缝地与所有其他指令一起操作，且设计为能获得最佳实时性能。在把两个 W 寄存器相乘时， MAC 类指令能够从存储器中同时取出两个操作数。为了能够同时取数据操作数，对于 MAC 类指令，数据空间拆分为两块，对所有其他指令数据空间则是线性的。对于 MAC 类指令，这是通过将某些工作寄存器专用于每个地址空间，以透明而灵活的方式实现的。

内核不支持多级指令流水线。不过，为了最大限度地获得可用的执行时间，使用了单级指令预取机制，在指令执行的前一个周期，存取并部分译码指令。除了一些特例外，大部分指令都在单个周期内执行。

内核具有用于陷阱和中断的向量异常处理结构，具有 62 个独立的向量。异常包括最多 8 个陷阱（其中有 4 个是保留的）和 54 个中断。根据用户指定的 1 到 7 之间的优先级 （1 为最低优先级， 7 为最高优先级），以及预定义的 "自然顺序"，决定每个中断的优先级。陷阱的优先级是固定的，其优先级范围是从 8 到 15。

2.2 编程模型

编程模型如图 2-1 所示，它包括 16 x 16 位工作寄存器（W0 至 W15）、2 x 40 位累加器（ACCA 和 ACCB）、状态寄存器（SR）、数据表页寄存器（TBLPAG）、程序空间可视性页寄存器（PSVPAG）、DO 和 REPEAT 寄存器（DOSTART、DOEND、DCOUNT 和 RCOUNT）以及程序计数器（PC）。工作寄存器可作为数据、地址或偏移量寄存器。全部寄存器都是存储器映射的。W0 用作文件寄存器寻址的 W 寄存器。

这些寄存器中，一些寄存器有与之关联的影子寄存器，如图 2-1 所示。影子寄存器用作临时保存寄存器，事件发生时可在影子寄存器和主寄存器之间传递内容。所有影子寄存器均不能直接存取。寄存器与其影子寄存器之间的数据传递遵守下列规则：

- `PUSH.S` 和 `POP.S`
  W0、W1、W2、W3 和 SR（仅限 DC、N、OV、Z 和 C 位）与对应的影子寄存器之间进行内容传递。
- `DO` 指令
  循环开始时，DOSTART、DOEND 和 DCOUNT 寄存器的内容压入影子寄存器，在循环结束时其内容从各自的影子寄存器中弹出。

对工作寄存器进行字节操作时，只影响目标寄存器的最低有效字节（LSB）。不过，存储器映射工作寄存器的一个好处是，可以通过字节宽数据存储空间存取来处理最低和最高有效字节。

2.2.1 软件堆栈指针 / 帧指针

dsPIC® DSC 器件具备一个软件堆栈。W15 被指定用作软件堆栈指针（Stack Pointer，SP），异常处理以及子程序调用与返回将自动修改 W15。但是，W15 可以被任何指令所引用，引用方式与引用所有其他 W 寄存器相同。这简化了堆栈指针的读、写和处理（例如，创建堆栈帧）。

注：	为了防止出现不对齐的堆栈访问，W15<0> 始终保持为零。

复位时 W15 初始化为 0x0800。在初始化过程中，用户可以重新设定 SP，使它指向数据空间内的任何单元。

W14 被指定用作 `LNK` 和 `ULNK` 指令中所定义的堆栈帧指针。但是，W14 可以被任何指令所引用，引用方式与引用所有其他 W 寄存器相同。

2.2.2 状态寄存器

dsPIC DSC 内核有一个 16 位状态寄存器（SR），其 LSB 称为 SR 低字节（SRL），其 MSB 称为 SR 高字节（SRH），SR 寄存器组成见图 2-1。

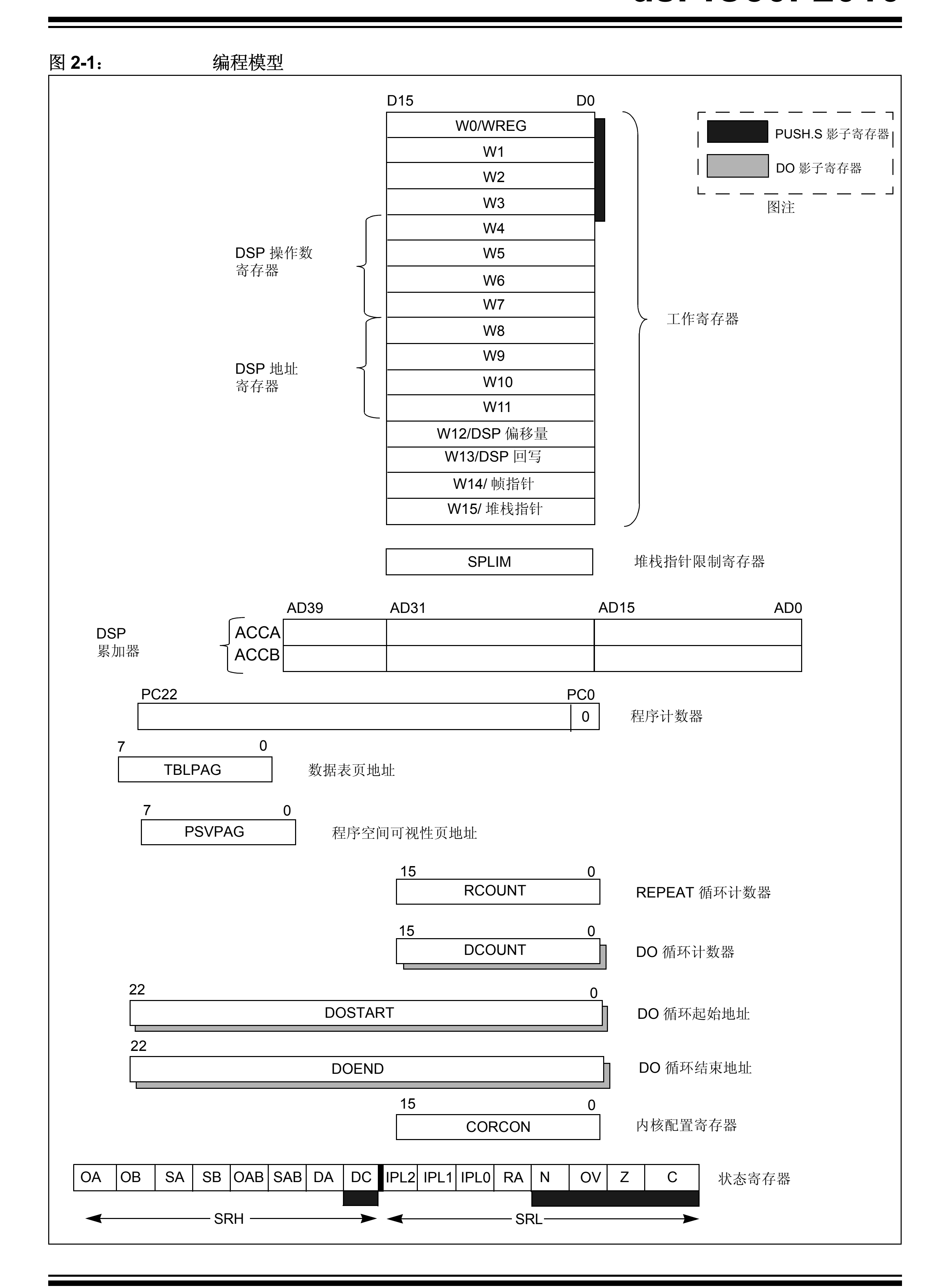
SRL 包含全部的 MCU ALU 运算状态标志位（包括 Z 位），以及 CPU 中断优先级状态位 IPL<2:0> 和 REPEAT 有效状态位 RA。在异常处理期间，SRL 与 PC 的 MSB 连接起来形成一个完整的字值，然后被压入堆栈。

状态寄存器的高字节包含 DSP 加法器 / 减法器状态位、DO 循环有效位（DA）和半进位（DC）状态位。

2.2.3 程序计数器

程序计数器为 23 位宽；bit 0 始终为零。因此，PC 能够最多寻址 4M 指令字。

图 **2-1**： 编程模型


D15
D0
W0/WREG
W1
W2
W3
W4
W5
W6
W7
W8
W9
W10
W11
W12/DSP 偏移量
W13/DSP 回写
W14/ 帧指针
W15/ 堆栈指针
DSP 操作数
寄存器
DSP 地址
寄存器
工作寄存器
PUSH.S 影子寄存器
DO 影子寄存器
图注
SPLIM
堆栈指针限制寄存器
AD39
AD31
AD15
AD0
DSP
累加器
ACCA
ACCB
PC22
PC0
0
程序计数器
7
0
TBLPAG
数据表页地址
7
0
PSVPAG
程序空间可视性页地址
15
0
RCOUNT
REPEAT 循环计数器
15
0
DCOUNT
DO 循环计数器
22
0
DOSTART
DO 循环起始地址
22
DOEND
DO 循环结束地址
15
0
CORCON
内核配置寄存器
OA
OB
SA
SB
OAB
SAB
DA
DC
IPL2
IPL1
IPL0
RA
N
OV
Z
C
状态寄存器
SRH
SRL

2.3 除法支持

dsPIC DSC 器件具有这样的特性：支持 16 位 /16 位有符号小数除法运算，支持 32 位 /16 位、 16 位 /16 位有符号和无符号整数除法运算，除法形式均为单指令迭代除法。支持以下指令和数据长度：

1. `DIVF` — 16/16 有符号小数除法
2. `DIV.sd` — 32/16 有符号除法
3. `DIV.ud` — 32/16 无符号除法
4. `DIV.sw` — 16/16 有符号除法
5. `DIV.uw` — 16/16 无符号除法

16/16 除法类似于 32/16 除法（它们的迭代次数相同），但在第一次迭代时，被除数要进行零扩展或符号扩展。

除法指令必须在一个 REPEAT 循环中执行。任何其他执行方式 （例如，一系列不连续的除法指令），将不会正常运行，因为指令流取决于 RCOUNT。除法指令不会自动设置 RCOUNT 值，于是 RCOUNT 必须显式地、正确地在 REPEAT 指令中给定，如表 2-1 所示（REPEAT 将执行目标指令 { 操作数的值 +1} 次）。必须为 DIV/DIVF指令的18次迭代设置REPEAT循环计数。所以，完整的除法操作需要 19 个周期。

注：	除法流可以被中断。然而，用户需要恰当地保存中断现场。

表 2-1： 除法指令

指令	功能
DIVF	有符号小数除法：Wm/Wn → W0 ； Rem → W1
DIV.sd	有符号除法：(Wm + 1:Wm)/Wn → W0 ； Rem → W1
DIV.ud	无符号除法：(Wm + 1:Wm)/Wn → W0 ； Rem → W1
DIV.sw（或 DIV.s）	有符号除法：Wm/Wn → W0 ； Rem → W1
DIV.uw（或 DIV.u）	无符号除法：Wm/Wn → W0 ； Rem → W1

2.4 DSP 引擎

DSP 引擎的组成：一个高速 17 位× 17 位乘法器、一个桶形移位寄存器和一个 40 位加法器 / 减法器 （两个目标累加器、舍入逻辑和饱和逻辑）。

DSP 引擎能够执行固有的不需要其他数据的累加器至累加器操作。这些指令是 ADD、 SUB 和 NEG。

通过 CPU 内核配置寄存器 （CORCON）中的各个位，可以有多种 DSP 引擎选择，这些选择如下：

1. 小数或整数 DSP 乘法 （IF）。
2. 有符号或无符号 DSP 乘法 （US）。
3. 常规或收敛舍入 （RND）。
4. ACCA 自动饱和使能 / 禁止 （SATA）。
5. ACCB 自动饱和使能 / 禁止 （SATB）。
6. 对于写数据存储器，自动饱和使能/禁止（SATDW）。
7. 累加器饱和模式选择 （ACCSAT）。

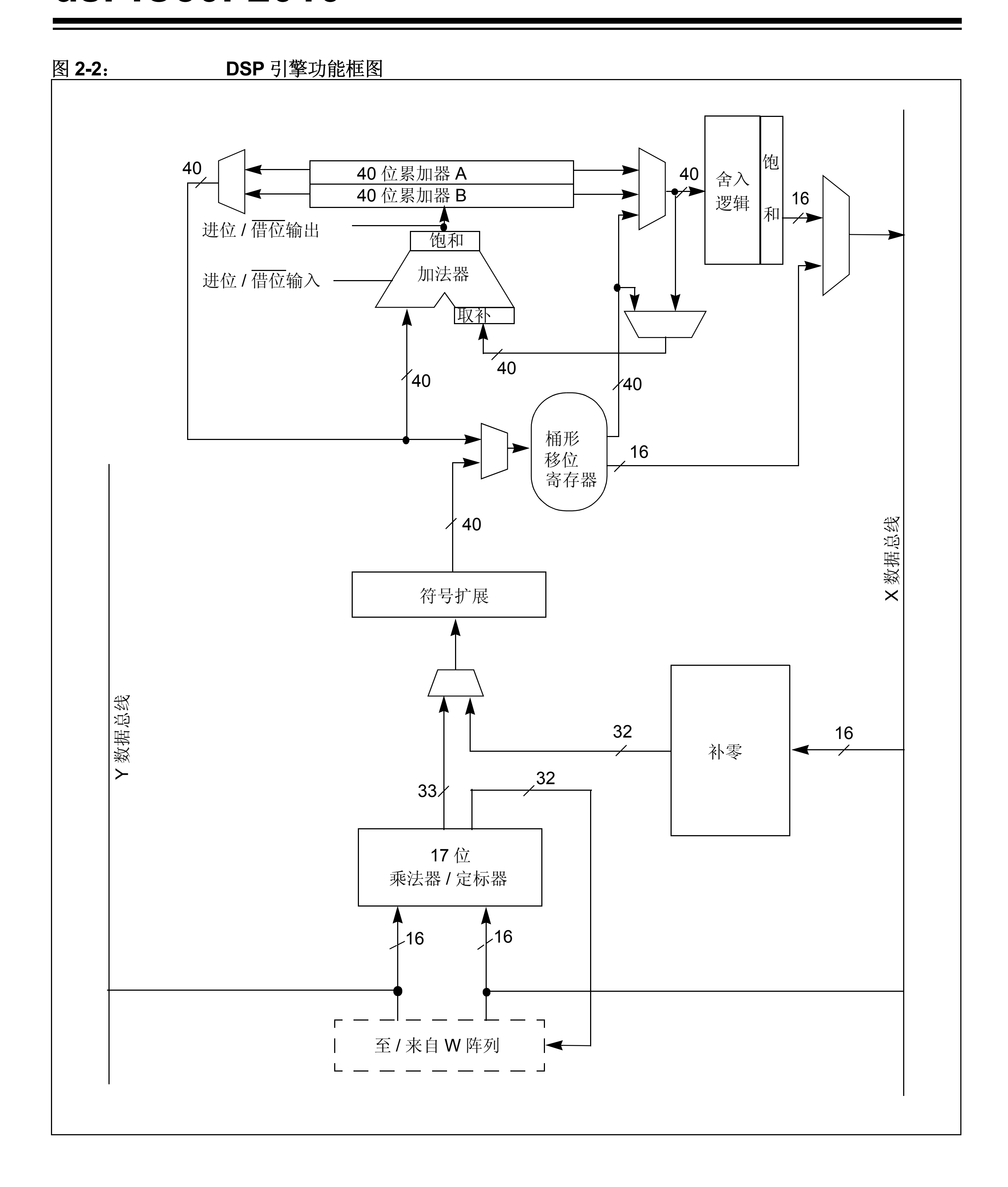
注： CORCON 的构成，见表 3-3。

DSP 引擎的框图，如图 2-2 所示。

表 **2-2**： **DSP** 指令汇总

指令	代数运算	**ACC** 回写？
CLR	A = 0	是
ED	$A = (x – y)^2$	否
EDAC	$A = A + (x – y)^2$	否
MAC	A = A + (x * y)	是
MAC	$A = A + x^2$	否
MOVSAC	A 中内容将不发生改变	是
MPY	A = x * y	否
MPY.N	A = – x * y	否
MSC	A = A – x * y	是

图 2-2： DSP 引擎功能框图



2.4.1 乘法器

17位×17位的乘法器可以进行有符号或无符号的运算，其输出经过定标器进行换算后可支持 1.31 小数（Q31）或 32 位整数结果。无符号操作数经过零扩展后，送入乘法器输入值的第 17 位。有符号操作数经过符号扩展，送入乘法器输入值的第 17 位。17 位× 17 位乘法器 / 定标器的输出是 33 位值，它将被符号扩展为 40 位。整型数据的固有表示形式为有符号的二进制补码值，其中，MSB 定义为符号位。一般来说，N 位二进制补码整数的范围为 -2^{N-1} 到 $2^{N-1} - 1$。对于 16 位整数，数据范围是 -32768 （0x8000）至 32767 （0x7FFF），包括 0 在内。对于 32 位整数，数据范围是 -2,147,483,648（0x8000 0000）至 2,147,483,645（0x7FFF FFFF）。

当乘法器配置为小数乘法时，数据表示为二进制补码小数，其中 MSB 定义为符号位，小数点暗含在符号位之后 （QX 格式）。暗含小数点的 N 位二进制补码小数的范围是 -1.0 至（$1 - 2^{1-N}$）。对于 16 位小数，Q15 数据范围是 -1.0 （0x8000）至 0.999969482 （0x7FFF），包括 0 在内，其精度为 3.01518×10^{-5}。在小数方式下，16x16 乘法运算将产生 1.31 乘积，其精度为 4.65661×10^{-10}。

同一个乘法器还用来支持 MCU 乘法指令，包括整数的 16 位有符号、无符号和混和符号乘法。

`MUL` 指令可以使用字节或字长度的操作数。字节操作数将产生 16 位结果，而字操作数将产生 32 位结果，结果存放在 W 寄存器阵列的指定寄存器中。

2.4.2 数据累加器和加法器 / 减法器

数据累加器包含一个 40 位的加法器 / 减法器，它带有自动符号扩展逻辑。它可以选择两个累加器 （A 或 B）之一作为其预累加的源累加器和后累加的目标累加器。对于 `ADD` 和 `LAC` 指令，将被累加或装入的数据可选择通过桶形移位寄存器在累加之前进行换算。

2.4.2.1 加法器 / 减法器，溢出和饱和

加法器 / 减法器是一个 40 位的加法器，一侧输入可以选择为零，而另一侧的输入可以是数据的原码或补码。对于加法，进位 / 借位输入是高有效的，另一侧输入是数据的原码 （没有求补的）；对于减法，进位 / 借位输入是低有效的，另一侧输入是数据的补码。溢出和饱和状态位 SA/SB 及 OA/OB 提供加法器 / 减法器的溢出和饱和状态信息，状态位被锁存在状态寄存器中并在其中得到反映。

- 从 bit 39 溢出：这是灾难性溢出，会破坏累加器的符号。
- 溢出到警戒位（bit 32 到 bit 39）：这是可恢复的溢出。每当警戒位有任何不一致，就将把这个状态位置 1。

加法器有一个额外的饱和块，如果选取的话，饱和模块将控制累加器的数据饱和。饱和模块使用加法器的结果、上述的溢出状态位以及 SATA/B （CORCON<7:6>）和 ACCSAT （CORCON<4>）模式控制位，来确定什么时候以及什么值要饱和。

在状态寄存器中有 6 个支持饱和及溢出的位，它们是：

1. OA：
   ACCA 溢出到警戒位
2. OB：
   ACCB 溢出到警戒位
3. SA：
   ACCA 已饱和 （bit 31 溢出并饱和）
   或者
   ACCA 溢出到警戒位并饱和（bit 39 溢出并饱和）
4. SB：
   ACCB 已饱和 （bit 31 溢出并饱和）
   或者
   ACCB 溢出到警戒位并饱和（bit 39 溢出并饱和）
5. OAB：
   OA 和 OB 的逻辑或 （OR）
6. SAB：
   SA 和 SB 的逻辑或 （OR）

每次数据通过加法器 / 减法器，就会修改 OA 和 OB 位。置 1 时，它们表明最近的操作已经溢出到累加器警戒位（bit 32 到 bit 39）。如果 OA 和 OB 位置 1 而且 INTCON1 寄存器中相应的溢出陷阱标志允许位 （OVATE 和 OVBTE）置 1 的话，还可以选择用 OA 和 OB 位产生算术警告陷阱 （见**第 5.0 节 “中断”**）。这使得用户能够立即采取措施，例如，改正系统增益。

每次数据通过加法器 / 减法器，就会修改 SA 和 SB 位，但用户只能对它们进行清零。置 1 时，它们表明累加器已经溢出其最大范围 （对于 32 位饱和是 bit 31，而 40 位饱和是 bit 39），将发生饱和（如果饱和使能的话）。如果没有使能饱和，SA 和 SB 置 1 默认为 bit 39 溢出，于是表明产生了灾难性溢出。如果 INTCON1 寄存器中的 COVTE 位置 1，当饱和被禁止时，SA 和 SB 位将产生算术警告陷阱。

在状态寄存器中，对于溢出和饱和状态位，可以将 OA 和 OB 的逻辑或形成 OAB 位，将 SA 和 SB 的逻辑或形成 SAB 位。这样，只需检查状态寄存器中的一个位，程序就能判断累加器是否已溢出，或者累加器是否已饱和。对于通常要使用两个累加器的复数运算而言，这很有用。

器件支持三种饱和及溢出模式：

1. bit 39 溢出和饱和：
   当发生 bit 39 溢出和饱和时，饱和逻辑将最大的正 9.31 值 （0x7FFFFFFFFF）或最大的负 9.31 值 （0x8000000000）装入目标累加器。 SA 或 SB 位置 1 并保持置 1 直到被用户清零。这称为 “超饱和” ，为错误数据或不可预期的算法问题（例如，增益计算）提供了保护机制。
2. bit 31 溢出和饱和：
   当发生 bit 31 溢出和饱和时，饱和逻辑将最大的正 1.31 值 （0x007FFFFFFF）或最大的负 1.31 值 （0x0080000000）装入目标累加器。 SA 或 SB 位置 1 并保持置 1 直到被用户清零。当这种饱和模式生效时，不使用警戒位（因此 OA、OB 或 OAB 位不会被置 1）。
3. bit 39 灾难性溢出：
   来自加法器的 bit 39 溢出状态位，用来将 SA 或 SB 位置 1（置 1 后，这两位将保持置 1，除非用户进行清零）。不进行饱和操作，允许累加器溢出（破坏其符号）。如果 INTCON1 寄存器中的 COVTE 位置 1，灾难性溢出会导致一个陷阱异常。

2.4.2.2 累加器 “回写”

MAC 类指令（MPY、MPY.N、ED 和 EDAC 除外）可以选择将累加器高字 （bit 16 至 bit 31）的舍入形式写入数据存储空间，前提是当前指令不对该累加器进行操作。通过 X 总线，寻址组合的 X 和 Y 地址空间，执行回写操作。支持以下寻址模式：

1. W13，寄存器直接寻址：
   非操作目标的累加器的舍入内容以 1.15 小数形式写入 W13。
2. [W13]+ = 2，执行后递增的寄存器间接寻址：
   非操作目标的累加器的舍入内容以 1.15 小数形式写入 W13 指向的地址。然后 W13 递增 2（对于字写入）。

2.4.2.3 舍入逻辑

舍入逻辑是一个组合块，在累加器写 （存储）过程中执行常规的 （有偏）或收敛的 （无偏）舍入功能。舍入模式由 CORCON 寄存器中 RND 位的状态决定。它会产生一个 16 位的 1.15 数据值，该值被送到数据空间写饱和逻辑：如果此指令不指明舍入，就会存储一个截取的 1.15 数据值，简单地丢弃低位字 （least significant word， lsw）。

常规舍入取累加器的 bit 15，对它进行零扩展并将扩展后的值加到 ACCxH 字 （累加器的 bit 16 至 bit 31）。如果 ACCxL 字 （累加器的 bit 0 至 bit 15）在 0x8000 和 0xFFFF 之间（包括 0x8000 ），则 ACCxH 递增 1。如果 ACCxL 在 0x0000 和 0x7FFF 之间，则 ACCxH 不变。此算法的结果经过一系列随机舍入操作，值将稍稍偏大 （正偏）。

除非 ACCxL 等于 0x8000，否则收敛的（或无偏）舍入操作方式与常规舍入相同。ACCxL 等于 0x8000 时，要对 ACCxH 的最低位 （累加器的 bit 16）进行检测。如果它为 1， ACCxH 递增 1。如果它为 0， ACCxH 不变。假设 bit 16 本身是随机的，这样的机制将消除任何可能累加的舍入偏移。

通过 X 总线， SAC 和 SAC.R 指令将目标累加器内容的截取（SAC）或舍入（SAC.R）形式存入数据存储空间（这受数据饱和的影响，见第 **2.4.2.4 节 “数据空间写饱和”**）。注意，对于 MAC 类指令，累加器回写操作将以同样的方式进行，通过 X 总线寻址组合的 MCU （X 和 Y）数据空间。对于 MAC 类指令，数据始终要进行舍入。

2.4.2.4 数据空间写饱和

除了加法器 / 减法器饱和，对数据空间进行写操作也会饱和，但不会影响源累加器的内容。数据空间写饱和逻辑块接受来自舍入逻辑块的一个 16 位的 1.15 小数值作为输入，还接受来自源 （累加器）和 16 位舍入加法器的溢出状态。这些经过组合，用来选择恰当的 1.15 小数值作为输出，写入至数据存储空间中。

如果 CORCON 寄存器中的 SATDW 位置 1，将检测（经过舍入或截取后的）数据是否溢出，并相应地进行调整。如果输入数据大于 0x007FFF，则写入数据存储器中的数据被强制为最大的正 1.15 值， 0x7FFF。如果输入数据小于 0xFF8000，则写入数据存储器中的数据被强制为最大的负 1.15 值， 0x8000。源累加器的最高位位 （bit 39）用来决定被检测的操作数的符号。

如果 CORCON 寄存器中的 SATDW 位没有置 1，则输入数据都将通过，在任何情况下都不会被修改。

2.4.3 桶形移位寄存器

桶形移位寄存器在单个周期内最多可算术或逻辑右移 16 位或左移 16 位。源寄存器可以是两个 DSP 累加器中的任何一个，或者是 X 总线（支持寄存器或存储器数据的多位移位）。

移位寄存器需要一个有符号二进制值，用来确定移位操作的幅度 （位数）和方向。正值将操作数右移。负值则将操作数左移。值为 “0”则不改变操作数。

桶形移位寄存器是 40 位宽的，于是，它为 DSP 移位操作提供了 40 位的结果，而为 MCU 移位操作提供 16 位的结果。来自 X 总线的数据在桶形移位寄存器中的存放方式是：右移则数据存放在 bit 16 至 bit 31，左移则存放在 bit 0 至 bit 15。

注：

3.0 存储器构成

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

3.1 程序地址空间

程序地址空间为 4M 指令字。它可通过一个 24 位值来寻址，这24位值来自23位PC、表指令有效地址（Effective Address，EA）或数据空间 EA（当按照表 3-1 定义的那样，将程序空间映射到数据空间时）。注意，为了提供与数据空间寻址的兼容性，在两个连续的程序字之间，程序空间地址递增量为 2。

对于除 TBLRD/TBLWT 外的所有访问，用户程序空间访问限制在低 4M 指令字地址范围（0x000000 至 0x7FFFFE）；TBLRD/TBLWT 使用TBLPAG<7>来决定用户或配置空间访问。在表 3-1 “程序空间地址构成”中，bit 23 允许访问器件 ID、用户 ID 和配置位。对于其他情形，bit 23 始终清零。

> 注： 图 3-1 所示的地址映射是概念性的，实际的存储器配置取决于可用的存储器大小，各器件的存储器配置可能会存在差异。

图 3-1： dsPIC30F2010 的程序存储空间映射

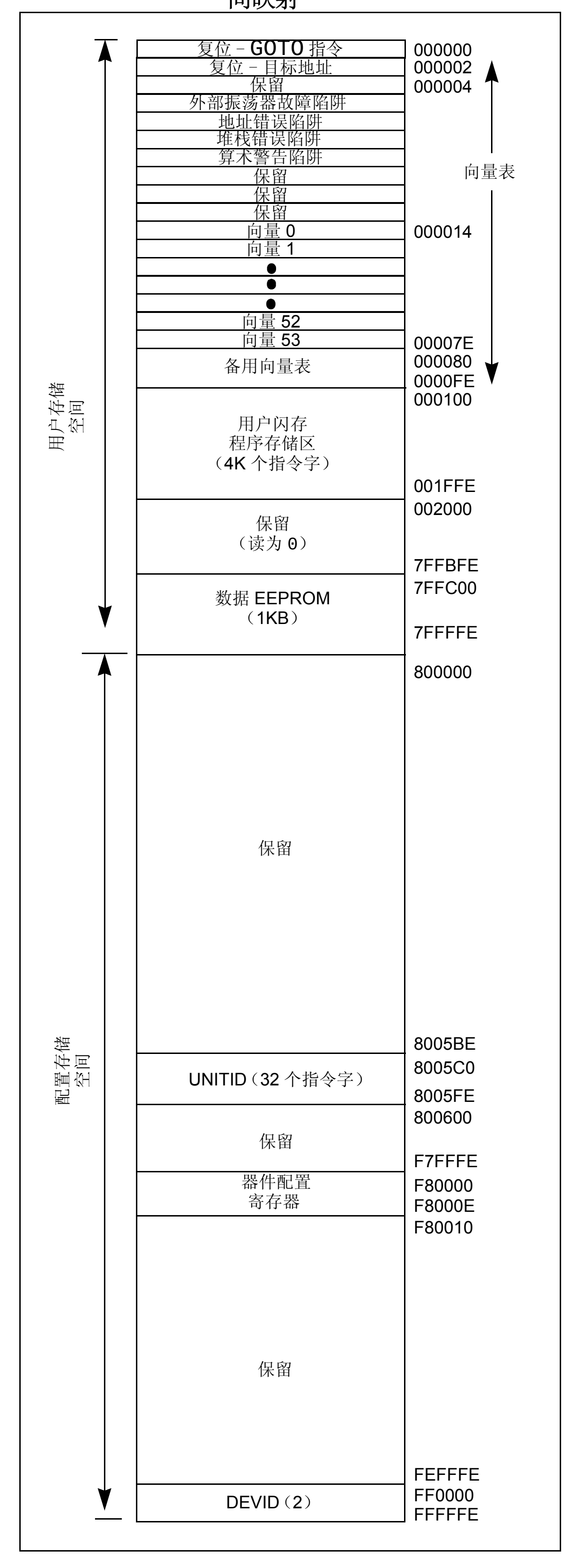
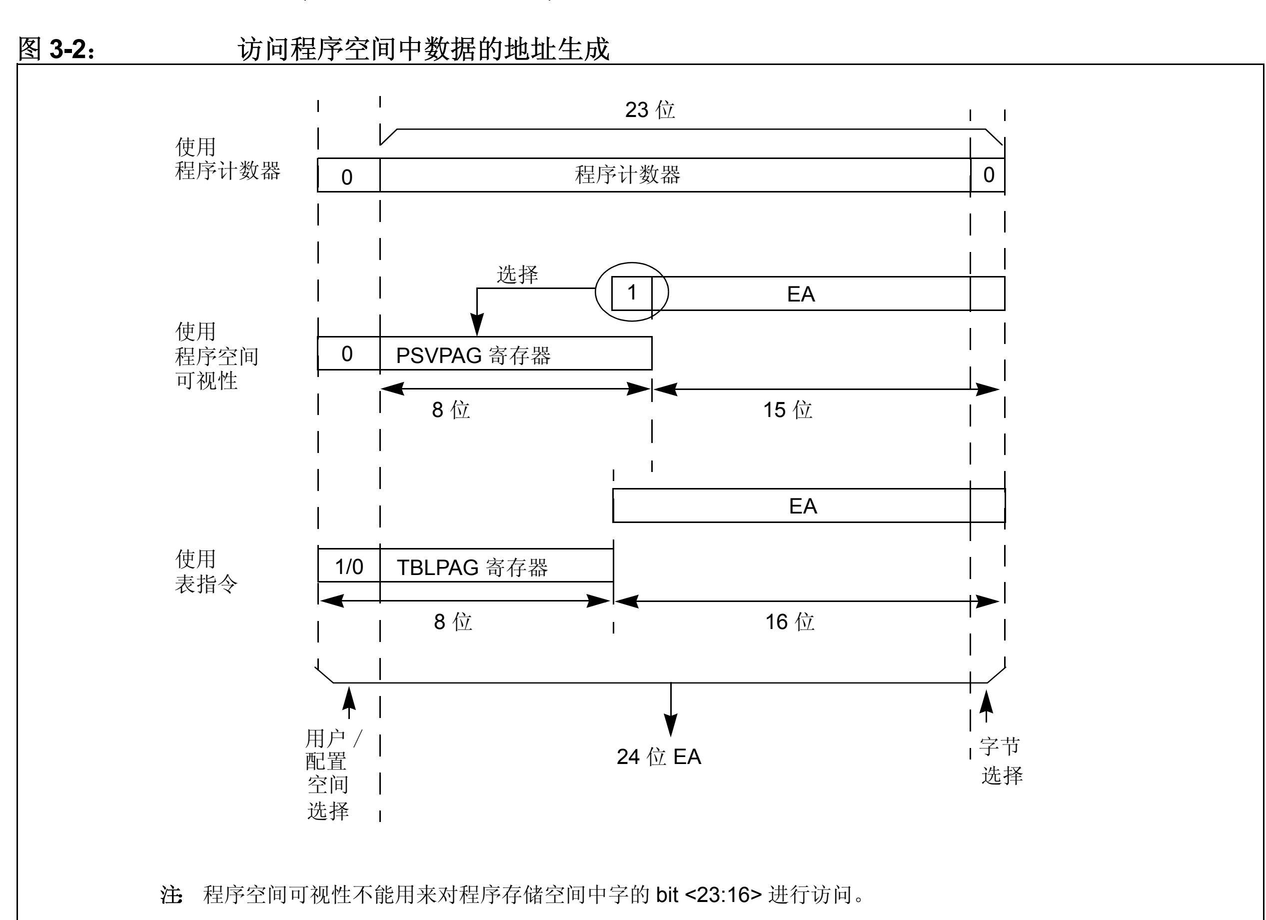


表 **3-1**： 程序空间地址构成

访问类型	访问空间	程序空间地址				
		<23>	**<22:16>**	**<15>**	**<14:1>**	**<0>**
指令访问	用户	0	PC<22:1>			0
TBLRD/TBLWT	用户（TBLPAG<7> = 0）	TBLPAG<7:0>		数据 EA <15:0>		
TBLRD/TBLWT	配置（TBLPAG<7> = 1）	TBLPAG<7:0>		数据 EA <15:0>		
程序空间可视性	用户	0	PSVPAG<7:0>		数据 EA <14:0>	

图 **3-2**： 访问程序空间中数据的地址生成



注： 程序空间可视性不能用来对程序存储空间中字的 bit <23:16> 进行访问。

3.1.1 使用表指令访问程序存储器中的数据

本器件架构取指令时是取 24 位宽的程序存储器。因此，指令始终是对齐的。不过，由于器件架构是改进型哈佛架构，数据也可以出现在程序空间中。

有两种方法可以访问程序空间：通过特殊的表指令，或者通过把 16K 字程序空间页重新映射到数据空间的上半部分（见**第 3.1.2 节 "使用程序空间可视性访问程序存储器中的数据"**）。TBLRDL 和 TBLWTL 指令，提供了读或写程序空间内任何地址的 lsw 的直接方法，无需通过数据空间。TBLRDH 和 TBLWTH 指令是可以把一个程序空间字的高 8 位作为数据存取的惟一方法。

对于每个连续的 24 位程序字，PC 的递增量为 2。这使得程序存储器地址能够直接映射到数据空间地址。于是，程序存储器可以看作是两个 16 位字宽的地址空间，它们并排放置，具有相同的地址范围。TBLRDL 和 TBLWTL 访问存有低位数据字的空间，而 TBLRDH 和 TBLWTH 则访问存有最高有效数据字节的空间。

图 3-2说明了如何为表操作和数据空间访问（PSV = 1）的创建 EA。这里，P<23:0> 指的是程序空间字，而 D<15:0> 指的是数据空间字。

器件提供了一组表指令，用来在程序空间和数据空间之间传送字节或字大小的数据。

1. TBLRDL：表读低位
   *字：*读程序地址的低位字； P<15:0> 映射到 D<15:0>；
   *字节：*读程序地址的 LSB 中的一个字节；
   当字节选择 = 0 时， P<7:0> 映射到目的字节；
   当字节选择 = 1 时， P<15:8> 映射到目的字节。
2. TBLWTL：表写低位字 （字节）（闪存编程的细节，见**第 6.0 节 "闪存程序存储器"**）。
3. TBLRDH：表读高位
   *字：*读程序地址的高位字； P<23:16> 映射到 D<7:0>； D<15:8> 始终 = 0。
   *字节：*读程序地址的 MSB 中的一个字节；
   当字节选择 = 0 时，P<23:16> 映射到目的字节；
   当字节选择 = 1 时，目的字节始终 = 0。
4. TBLWTH：表写高位字 （字节）（闪存编程的细节，见**第 6.0 节 "闪存程序存储器"**）。

图 3-3： 程序数据表访问 （低位字）

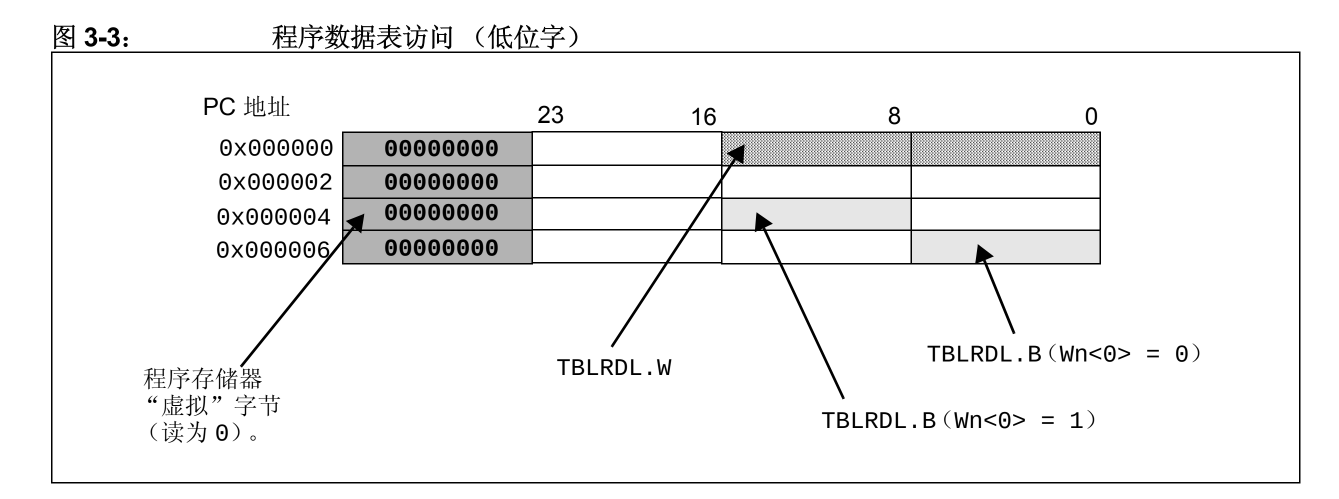
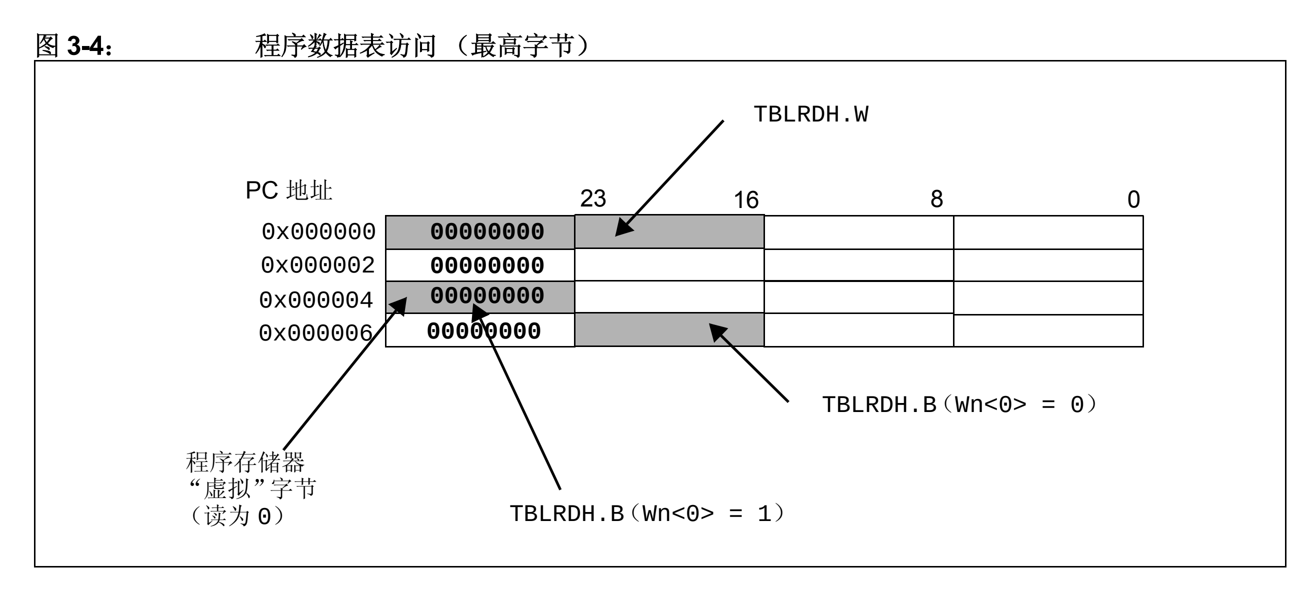


图 3-4： 程序数据表访问 （最高字节）



3.1.2 使用程序空间可视性访问程序存储器中的数据

可选择将数据空间的高 32 KB 映射到任何 16K 字程序空间页。这提供了通过 X 数据空间对存储的常量数据的透明访问，而无需使用特殊指令（即 TBLRDL/H 和 TBLWTL/H 指令）。

如果数据空间 EA 的 MSb 置 1，并且程序空间可视性使能 （方法是在内核控制寄存器 CORCON 中把 PSV 位置 1）时，就能通过数据空间访问程序空间。CORCON 功能的讨论请参见**第 2.4 节 “DSP 引擎”**。

正在执行的指令，如果要对这个区域进行数据访问的话，就需要一个额外的指令周期，因为需要进行两次程序空间读取操作。

注意，可寻址数据空间的上半部分始终是 X 数据空间的一部分。于是，当 DSP 操作使用程序空间映射来访问这个存储区域时， Y 数据空间通常应该存放 DSP 操作的状态（可变）数据，而 X 数据空间通常应该存放系数（常量）数据。

尽管每个数据空间地址， 0x8000 和更高，直接映射到对应的程序存储器地址（见图 3-6），但只使用 24 位程序字的低 16 位来存放数据。应该进行恰当的设置，使得对高 8 位进行访问的指令都成为非法指令，以维持器件的可靠性。指令编码的细节，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

注意，对于每个程序存储字 PC 都将递增 2，数据空间地址的低 15 位将直接映射到相应程序空间地址的低 15 位。剩下的位由程序空间可视性页寄存器（PSVPAG<7:0>）提供，如图 3-5 所示。

注：	在表读 / 写期间，暂时禁止 PSV 访问。

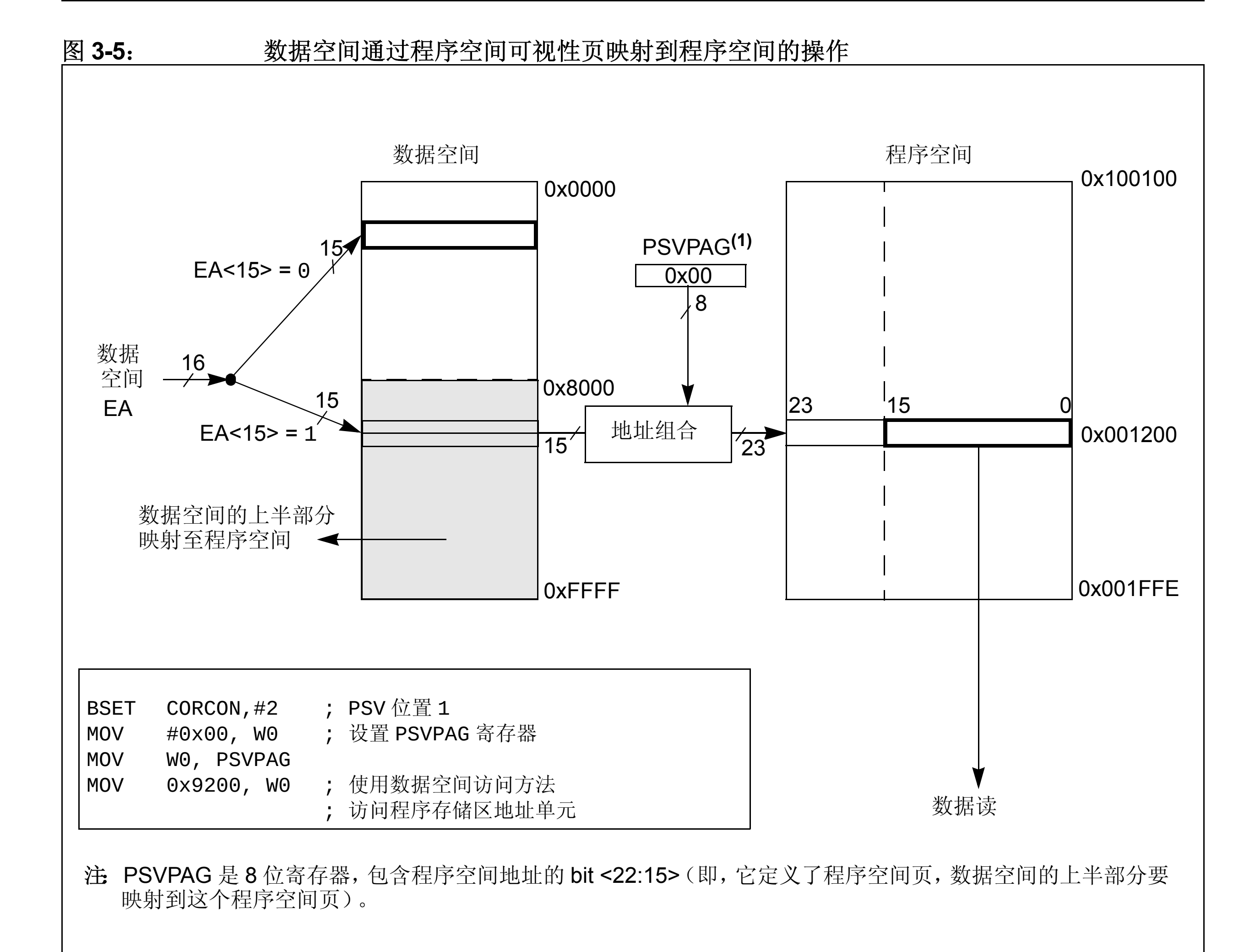
对于使用 PSV 而又在 REPEAT 循环之外执行的指令：

- 下列指令，除了规定的执行时间之外，需要一个额外的指令周期：
  - 带数据操作数预取的 MAC 类指令
  - MOV 指令
  - MOV.D 指令
- 其他所有的指令，除了规定的指令执行时间之外，需要两个额外的指令周期。

对于使用 PSV 而又在 REPEAT 循环内执行的指令：

- 下列情况，除了规定的指令执行时间之外，需要两个额外的指令周期：
  - 在第一次迭代中执行的指令
  - 在最后一次迭代中执行的指令
  - 由于中断而退出循环之前执行的指令
  - 中断得到处理后而再次进入循环时执行的指令
- REPEAT 循环的所有其他各次迭代，都允许使用 PSV 访问数据的指令在一个周期内执行。

图 3-5： 数据空间通过程序空间可视性页映射到程序空间的操作




```
BSET   CORCON,#2     ; PSV 位置 1
MOV    #0x00, W0     ; 设置 PSVPAG 寄存器
MOV    W0, PSVPAG
MOV    0x9200, W0    ; 使用数据空间访问方法
                     ; 访问程序存储区地址单元
```

注： PSVPAG 是 8 位寄存器，包含程序空间地址的 bit <22:15>（即，它定义了程序空间页，数据空间的上半部分要映射到这个程序空间页）。

3.2 数据地址空间

内核具有两个数据空间。数据空间可以看作是独立的（对于某些 DSP 指令而言），或者看作是统一的线性地址范围（对于 MCU 指令而言）。使用两个地址发生单元（Address Generation Units，AGU）和独立的数据路径访问数据空间。

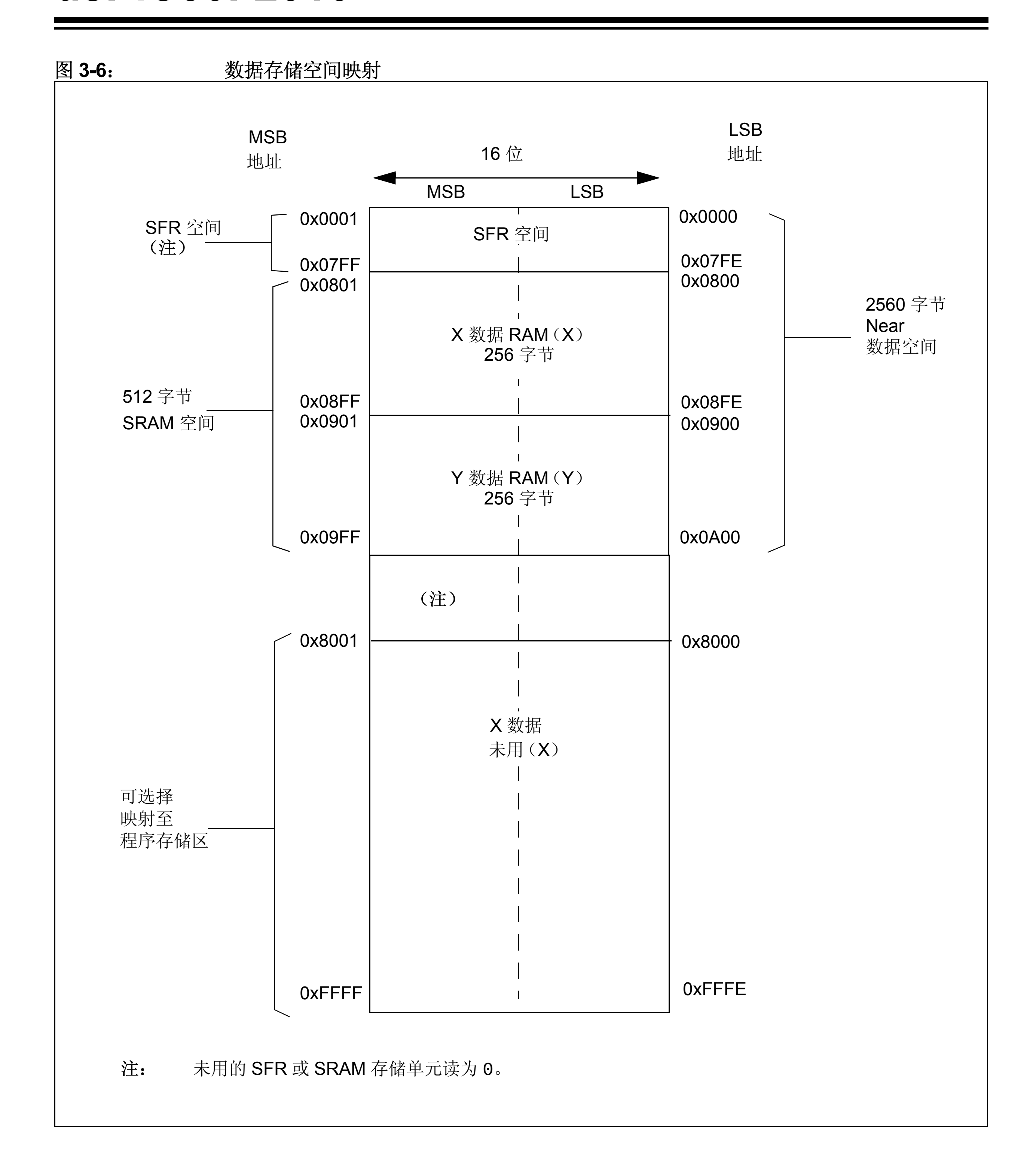
3.2.1 数据存储空间映射

数据存储空间分为两块，X 和 Y 数据空间。这个架构的关键之处在于，Y 空间是 X 空间的子集，Y 空间完全包含在 X 中。为了提供外在的线性寻址空间，X 和 Y 空间要有连续的地址。

在执行除 MAC 类指令之外的任何指令时， X 块由 256 字节数据地址空间（包括全部 Y 地址）组成。在执行一条 MAC 类指令时，X 块由 256 字节数据地址空间组成，但不包括 Y 地址块 （只用来读数据）。换句话说，所有其他指令把整个数据存储空间当作一个复合的地址空间。MAC 类指令把 Y 地址空间从数据空间中抽取出来，使用源自 W10 和 W11 的 EA 来寻址 Y 地址空间。剩下的 X 数据空间则使用 W8 和 W9 来寻址。只有 MAC 类指令才能同时访问两个地址空间。

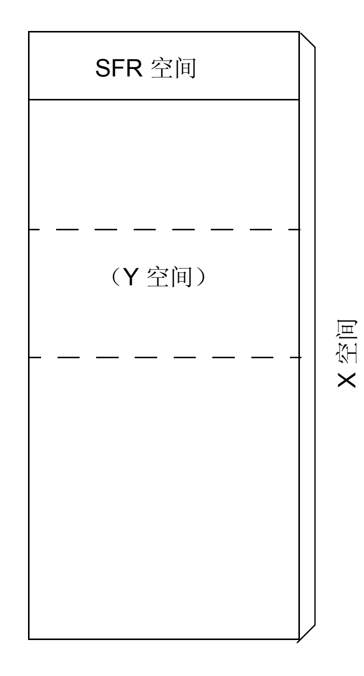
数据存储空间映射如图 3-6 所示。

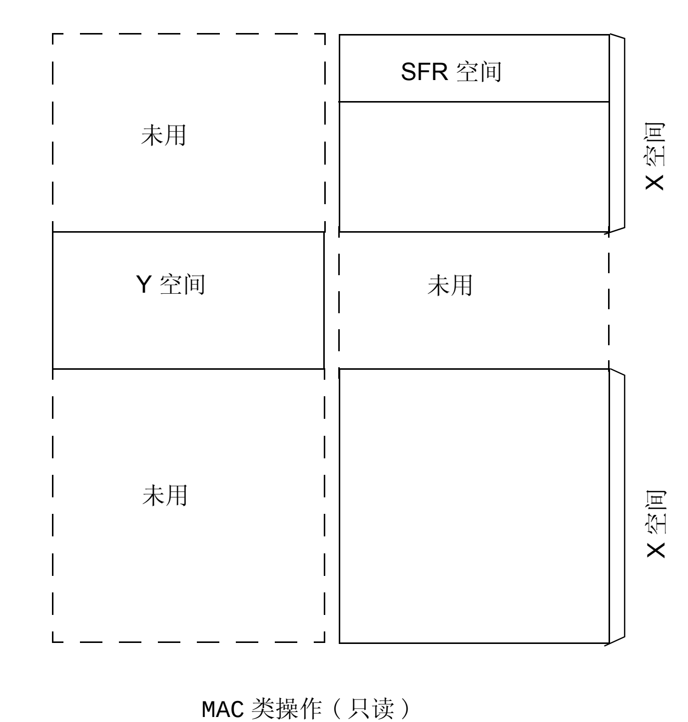
图 3-6： 数据存储空间映射



注： 未用的 SFR 或 SRAM 存储单元读为 0。

图 3-7： 用于 MCU 和 DSP （MAC 类）指令的数据空间





非 MAC 类操作（读 / 写）
MAC 类操作（写）

MAC 类操作（只读）

使用任何 W 寄存器的间接 EA

使用 W8 和 W9 的间接 EA　使用 W10 和 W11 的间接 EA

3.2.2 数据空间

X 数据空间为全部指令所使用，支持全部的寻址模式。读数据总线和写数据总线是独立的。X 读数据总线是所有指令的返回数据路径，它把数据空间视为复合的 X 和 Y 地址空间。对于双操作数读指令（MAC 类），它还是 X 地址空间的数据路径。对于所有指令而言， X 写数据总线是至数据空间的只写路径。

X 数据空间还支持所有指令的模寻址，不过要受到寻址模式的限制。位反转寻址只是在写 X 数据空间时才支持。

MAC 类指令（CLR、ED、EDAC、MAC、MOVSAC、MPY、MPY.N 和 MSC）使用的 Y 数据空间，协同 X 数据空间，提供了两条并行的数据读取路径。Y 总线上不会出现写操作。MAC 类指令使用两个专用的 W 寄存器指针， W10 和 W11，它们始终寻址 Y 数据空间，与 X 数据空间独立；而 W8 和 W9 则始终寻址 X 数据空间。注意，在累加器回写期间，数据地址空间视为 X 和 Y 数据空间的组合，从而写操作通过 X 总线进行。所以，可以写整个数据空间中的任何地址。

Y 数据空间仅用于与 MAC 类指令关联的数据预取操作。它也支持自动循环缓冲区的模寻址。当然，所有其他指令可以通过 X 数据路径来把 Y 数据地址空间作为复合线性空间的一部分来进行访问。

图 3-6 给出了 X 和 Y 数据空间之间的边界定义，用户不能设定改变边界。如果 EA 指向其所在地址空间之外的数据，或者指向物理存储器之外的存储单元，将返回全零的字 / 字节。例如，尽管 Y 地址空间对于使用任何寻址模式的所有非 MAC 类指令而言，是可见的，如果一条 MAC 类指令试图使用 W8 或 W9（X 空间指针）从 Y 地址空间取数据，将返回 0x0000。

表 3-2： 非法存储器访问的后果

试图进行的操作	返回数据
EA = 未实现地址	0x0000
在 MAC 类指令中使用 W8 或 W9 访问 Y 数据空间	0x0000
在 MAC 类指令中使用 W10 或 W11 访问 X 数据空间	0x0000

所有有效地址都是 16 位宽的，且指向数据空间内的字节。所以，数据空间地址范围是 64KB 或 32K 字。

3.2.3 数据空间宽度

内核数据宽度为 16 位。所有内部寄存器都按照 16 位宽的字来组织。数据存储空间以字节可寻址的 16 位宽的块来组织。

3.2.4 数据对齐

为了维持与 PIC® MCU 器件的向后兼容性，并提高数据存储空间的使用效率，dsPIC30F 指令集既支持字操作，也支持字节操作。在数据存储器和寄存器中，数据按照字来对齐，但是，所有数据空间有效地址都将解析为字节。数据字节读取将读取包含字节的整个字，使用任何有效地址 （EA）的 LSb 确定要选取的字节。选取的字节将存放到 X 数据路径的 LSB 中 （不可能出现来自 Y 数据路径的字节访问，因为 MAC 类指令只能取整字）。这就是说，数据存储器和寄存器组织为两个并行的字节宽的实体，它们共享 （字）地址译码，但写入线独立。数据字节写，将只写阵列或寄存器中与字节地址匹配的那一侧。

这种字节可访问能力的结果是，所有的有效地址计算（包括那些由 DSP 操作产生的有效地址，它们只能是字长度的数据）在内部进行换算，以适应字对齐的存储空间。例如，内核将识别执行后修改寄存器间接寻址模式 [Ws++] 的结果，对于字节操作产生的值是 Ws + 1，而对于字操作产生的值是 Ws + 2。

所有的字访问必须按照偶数地址对齐。不支持没有对齐的字数据取，所以在混和字节和字操作时，或者从 8 位 MCU 代码进行转换时，必须要小心。如果试图进行未对齐的读或写，将产生地址错误陷阱。如果在读操作时产生错误，正在执行的指令将完成，而如果在写操作时产生错误，指令仍将执行，但不会进行写入。无论是哪种情况，都将产生陷阱，从而系统和 / 或用户能够检查地址错误发生之前的机器状态。

图 3-8： 数据对齐

	MSB (15 … 8)	LSB (7 … 0)	
0001	字节 1	字节 0	0000
0003	字节 3	字节 2	0002
0005	字节 5	字节 4	0004

所有载入 W 寄存器的字节都将载入 W 寄存器的 LSB，W 寄存器的 MSB 不变。

提供了一条符号扩展（SE）指令，允许用户把 8 位的有符号数据转换为 16 位有符号值。或者，对于 16 位无符号数据，用户可以清零任何 W 寄存器的 MSB，方法是在相应的地址处执行一条零扩展（ZE）指令。

尽管大多数指令能够对字或字节大小的数据进行操作，但应该注意的是，一些指令，包括 DSP 指令，只对字大小的数据进行操作。

3.2.5 NEAR 数据空间

X 地址存储空间中，在 0x0000 和 0x1FFF 之间保留了一个 8KB 的 near 数据空间；在所有的存储器直接寻址指令中，可以通过一个 13 位的绝对地址来直接访问这个数据空间。其余的 X 地址空间和全部的 Y 地址空间都是间接可寻址的。此外，使用 MOV 指令可以寻址整个 X 数据空间，这支持通过 16 位地址字段进行存储器直接寻址。

3.2.6 软件堆栈

dsPIC DSC 器件具备一个软件堆栈。 W15 用作堆栈指针。

堆栈指针总是指向堆栈顶部第一个可供使用的字，从低地址到高地址方向增长。堆栈指针在弹出堆栈之前递减，而在压入堆栈后递增，如图 3-9 所示。注意，对于任何 CALL 指令时的 PC 压栈，在压入堆栈之前， PC 的MSB要进行零扩展，从而确保了MSB始终是清零的。

注：	在异常处理期间，在 PC 压入堆栈之前，要先将PC 的MSB与SRL寄存器组合在一起。

堆栈指针限制寄存器（SPLIM）与堆栈指针相关联。复位时 SPLIM 未被初始化。与堆栈指针的情况一样，SPLIM<0> 强制为 0，因为所有的堆栈操作必须是字对齐的。每当使用 W15 作为源指针或目的指针产生 EA 时，将与 SPLIM 中的值进行比较。如果堆栈指针（W15）与 SPLIM 寄存器的内容相等，则执行压入操作，不产生堆栈错误陷阱。但在随后的压栈操作时将会产生堆栈错误陷阱。这样的话，当堆栈增长超过 RAM 中地址 0x2000 时，如果要想产生堆栈错误陷阱，用值 0x1FFE 来初始化 SPLIM 即可。

类似地，当堆栈指针地址小于 0x0800 时，就会产生堆栈指针下溢（堆栈错误）陷阱，这避免了堆栈干预特殊功能寄存器（SFR）空间。

在对 SPLIM 寄存器进行写操作之后，不应紧跟着使用 W15 进行间接读操作的指令。

图 3-9： CALL 堆栈帧

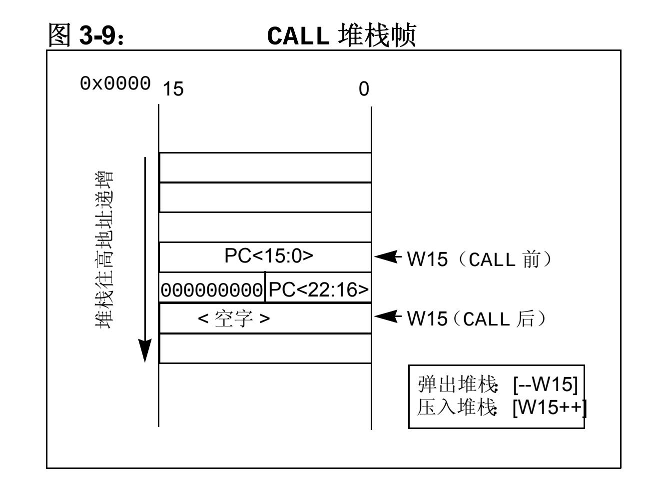


表 3-3： 内核寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
W0	0000	W0/WREG																0000 0000 0000 0000
W1	0002	W1																0000 0000 0000 0000
W2	0004	W2																0000 0000 0000 0000
W3	0006	W3																0000 0000 0000 0000
W4	0008	W4																0000 0000 0000 0000
W5	000A	W5																0000 0000 0000 0000
W6	000C	W6																0000 0000 0000 0000
W7	000E	W7																0000 0000 0000 0000
W8	0010	W8																0000 0000 0000 0000
W9	0012	W9																0000 0000 0000 0000
W10	0014	W10																0000 0000 0000 0000
W11	0016	W11																0000 0000 0000 0000
W12	0018	W12																0000 0000 0000 0000
W13	001A	W13																0000 0000 0000 0000
W14	001C	W14																0000 0000 0000 0000
W15	001E	W15																0000 1000 0000 0000
SPLIM	0020	SPLIM																0000 0000 0000 0000
ACCAL	0022	ACCAL																0000 0000 0000 0000
ACCAH	0024	ACCAH																0000 0000 0000 0000
ACCAU	0026	符号扩展 （ACCA<39>）								ACCAU								0000 0000 0000 0000
ACCBL	0028	ACCBL																0000 0000 0000 0000
ACCBH	002A	ACCBH																0000 0000 0000 0000
ACCBU	002C	符号扩展 （ACCB<39>）								ACCBU								0000 0000 0000 0000
PCL	002E	PCL																0000 0000 0000 0000
PCH	0030	—	—	—	—	—	—	—	—	—	PCH							0000 0000 0000 0000
TBLPAG	0032	—	—	—	—	—	—	—	—	TBLPAG								0000 0000 0000 0000
PSVPAG	0034	—	—	—	—	—	—	—	—	PSVPAG								0000 0000 0000 0000
RCOUNT	0036	RCOUNT																uuuu uuuu uuuu uuuu
DCOUNT	0038	DCOUNT																uuuu uuuu uuuu uuuu
DOSTARTL	003A	DOSTARTL															0	uuuu uuuu uuuu uuu0
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	DOSTARTH							0000 0000 0uuu uuuu
DOENDL	003E	DOENDL															0	uuuu uuuu uuuu uuu0
DOENDH	0040	—	—	—	—	—	—	—	—	—	DOENDH							0000 0000 0uuu uuuu
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

表 3-3： 内核寄存器映射 （续）

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
CORCON	0044	—	—	—	US	EDT	DL2	DL1	DL0	SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0000 0000 0010 0000
MODCON	0046	XMODEN	YMODEN	—	—	BWM<3:0>				YWM<3:0>				XWM<3:0>				0000 0000 0000 0000
XMODSRT	0048	XS<15:1>															0	uuuu uuuu uuuu uuu0
XMODEND	004A	XE<15:1>															1	uuuu uuuu uuuu uuu1
YMODSRT	004C	YS<15:1>															0	uuuu uuuu uuuu uuu0
YMODEND	004E	YE<15:1>															1	uuuu uuuu uuuu uuu1
XBREV	0050	BREN	XB<14:0>															uuuu uuuu uuuu uuuu
DISICNT	0052	—	—	DISICNT<13:0>														0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

4.0 地址发生器单元

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

dsPIC DSC 内核包含两个独立的地址发生器单元：X AGU 和 Y AGU。Y AGU 仅支持 DSP MAC 类指令的字数据读取。dsPIC DSC AGU 支持如下三种数据寻址类型：

- 线性寻址
- 模（循环）寻址
- 位反转寻址

线性和模数据寻址模式可应用于数据空间或程序空间。位反转寻址只能用于数据空间地址。

4.1 指令寻址模式

寻址模式经过优化可以支持各指令的具体功能，基本的寻址模式在表 4-1 中给出。MAC 类指令中提供的寻址模式，与其他指令类型中的寻址模式略有不同。

4.1.1 文件寄存器指令

大多数文件寄存器指令使用一个 13 位地址字段（f）来直接寻址数据存储器中的前 8192 字节（near 数据空间）。大多数文件寄存器指令使用工作寄存器 W0，W0 在这些指令中表示为 WREG。目的寄存器通常是同一个文件寄存器或者 WREG（MUL 指令除外），把结果写入寄存器或寄存器对。使用 MOV 指令能够获得更大的灵活性，可以访问整个数据空间。

表 4-1： 支持的基本寻址模式

寻址模式	说明
文件寄存器直接寻址	明确指定文件寄存器的地址。
寄存器直接寻址	直接访问寄存器的内容。
寄存器间接寻址	Wn 的内容形成有效地址（EA）。
执行后修改的寄存器间接寻址	Wn 的内容形成 EA。然后用一个常量值来修改 Wn（递增或递减）。
执行前修改的寄存器间接寻址	先用一个有符号常量值修改 Wn（递增或递减），Wn 的内容形成 EA。
寄存器偏移量的寄存器间接寻址	Wn 和 Wb 的和形成 EA。
立即数偏移量的寄存器间接寻址	Wn 和立即数的和形成 EA。

4.1.2 MCU 指令

三操作数 MCU 指令的形式是：

操作数 3 = 操作数 1 < 功能 > 操作数 2

其中，操作数 1 始终是工作寄存器（即，寻址模式只能是寄存器直接寻址），称为 Wb。操作数 2 可以是一个 W 寄存器，取自数据存储器或一个 5 位立即数。结果位置（操作数 3）可以是 W 寄存器或地址单元。MCU 指令支持下列寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 5 位或 10 位立即数寻址

> **注：** 并非所有的指令都支持上面给出的全部寻址模式。各条指令可能支持这些寻址模式的某些模式，指令不同支持的寻址模式可能不同。

4.1.3 传送指令和累加器指令

与其他指令相比，传送和 DSP 累加器类指令提供了更为零活的寻址模式。除了大多数 MCU 指令支持的寻址模式以外，传送和累加器指令还支持寄存器偏移量的寄存器间接寻址模式，这也叫做寄存器变址寻址模式。

> **注：** 对于 MOV 指令，指令中指定的寻址模式对于源寄存器和目的寄存器有效地址，可以是不同的。然而， 4 位 Wb （寄存器偏移量）字段为源寄存器和目寄存器所共用（但通常只由源寄存器或目的寄存器之一使用）。

概括地说，传送指令和累加器指令支持下列寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 寄存器偏移量的寄存器间接寻址 （变址寻址）
- 立即数偏移量的寄存器间接寻址
- 8 位立即数寻址
- 16 位立即数寻址

> **注：** 并非所有的指令都支持上面给出的全部寻址模式。各条指令可能支持这些寻址模式的某些模式，指令不同支持的寻址模式可能不同。

4.1.4 MAC 类指令

双源操作数 DSP 指令（CLR、ED、EDAC、MAC、MPY、MPY.N、MOVSAC 和 MSC），也叫做 MAC 类指令，它们使用一组简化的寻址模式，允许用户通过寄存器间接寻址表高效地对数据指针进行操作。

双源操作数预取寄存器必须是集合 {W8, W9, W10, W11} 的成员。对于数据读取，W8 和 W9 总是分配给 X RAGU，而 W10 和 W11 则始终用于 Y AGU。从而，产生的有效地址 （无论是在修改之前还是之后），对于 W8 和 W9 必须是 X 数据空间中的有效地址，对于 W10 和 W11 则必须是 Y 数据空间中的有效地址。

> **注：** 带寄存器偏移量的寄存器间接寻址，仅可用于 W9 （在 X 空间中）和 W11 （在 Y 空间中）。

概括地说， MAC 类指令支持下列寻址模式：

- 寄存器间接寻址
- 执行后修改 （修改量为 2）的寄存器间接寻址
- 执行后修改 （修改量为 4）的寄存器间接寻址
- 执行后修改 （修改量为 6）的寄存器间接寻址
- 带寄存器偏移量的寄存器间接寻址 （变址寻址）

4.1.5 其他指令

除了上述的各种寻址模式之外，一些指令使用各种大小的立即数。例如， BRA （转移）指令使用 16 位有符号立即数来直接指定转移的目标，而 DISI 指令使用一个 14 位无符号立即数字段。在一些指令中，比如 ADD Acc，操作数的来源和运算结果已经暗含在操作码中。某些操作，比如 NOP，没有任何操作数。

4.2 模寻址

模寻址模式，是一种使用硬件来自动支持循环数据缓冲区的方法。目的是在执行紧密循环代码时 （这在许多 DSP 算法中很典型），不需要用软件来执行数据地址边界检查。

可以在数据空间或程序空间中进行模寻址（因为这两种空间的数据指针机制本质上是相同的）。每个 X （也提供指向程序空间的指针）和 Y 数据空间中都可支持一个循环缓冲区。模寻址可以对任何 W 寄存器指针进行操作。然而，最好不要将 W14 或 W15 用于模寻址，因为这两个寄存器分别用作堆栈帧指针和堆栈指针。

总的来说，任何特定的循环缓冲区只能配置为单向工作，因为根据缓冲区的方向，对缓冲区起始地址（对于递增缓冲区）或结束地址（对于递减缓冲区）有某些限制。

使用限制的唯一例外是那些长度为 2 的幂的缓冲区。这些缓冲区满足起始和结束地址判据，它们可以双向工作（即，在低地址边界和高地址边界上都将进行地址边界检查）。

4.2.1 起始地址和结束地址

模寻址机制要求指定起始和结束地址，并把它们载入 16 位模缓冲区地址寄存器 （XMODSRT、 XMODEND、 YMODSRT 和 YMODEND，见表 3-3）中。

注：	Y 空间模寻址的 EA 计算使用字长度的数据（每个 EA 的 LSb 始终清零）。

循环缓冲区的长度没有直接指定，相应的起始地址、结束地址之差可以确定长度。循环缓冲区最大长度为 32K 字 （64KB）。

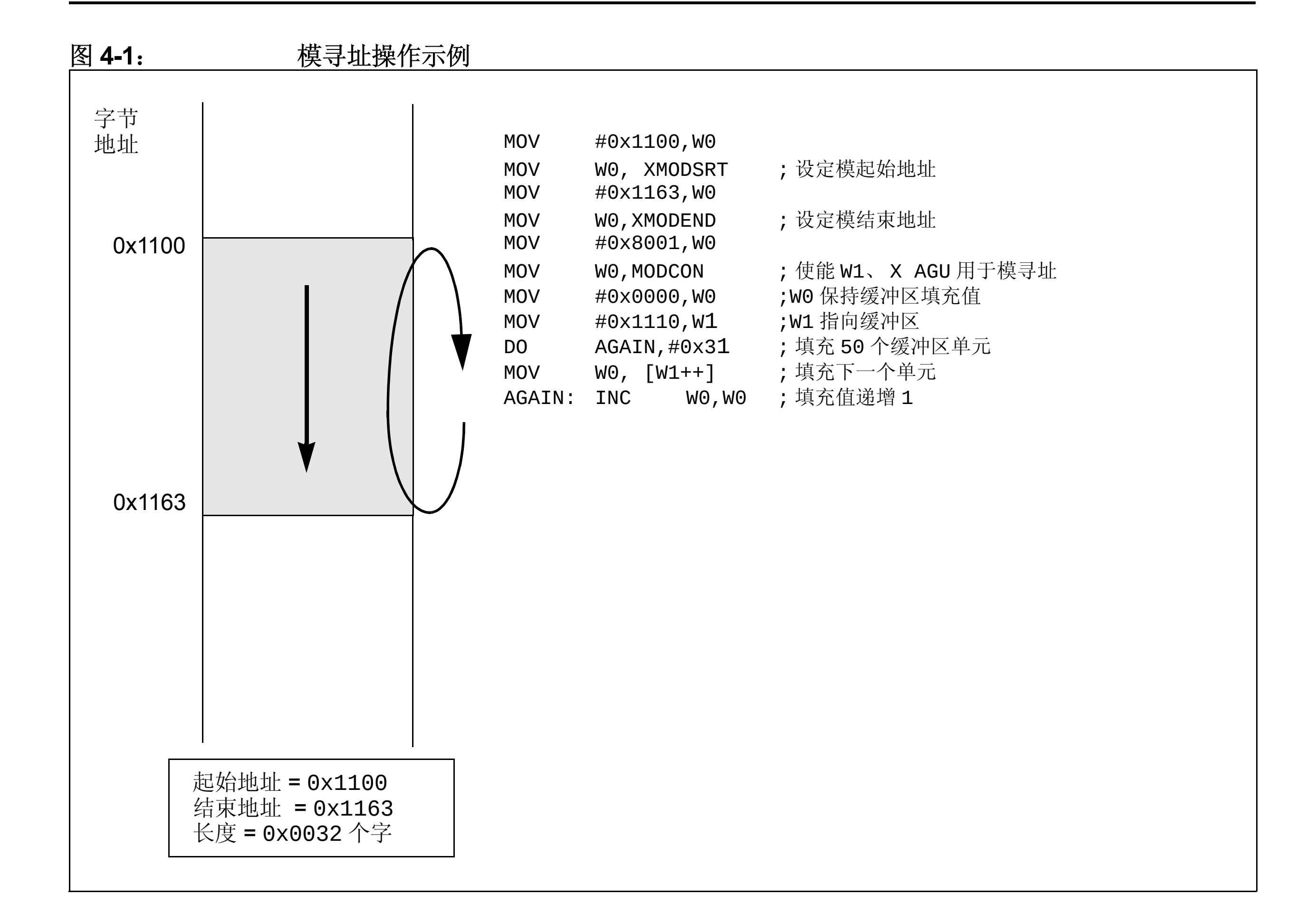
4.2.2 W 地址寄存器选择

模寻址和位反转寻址控制寄存器 MODCON<15:0> 中包含使能标志以及指定 W 地址寄存器的 W 寄存器字段。XWM 和 YWM 字段选择对哪些寄存器进行模寻址。如果 XWM = 15，则禁止 X RAGU 和 X WAGU 模寻址。类似地，如果 YWM = 15，禁止 Y AGU 模寻址。

要对其进行模寻址的 X 地址空间指针 W 寄存器（XWM）位于 MODCON<3:0> 中（见表 3-3）。当 XWM 被设置为除 15 之外的任何值且 XMODEN 位（MODCON<15>）置 1 时， X 数据空间的模寻址被使能。

要对其进行模寻址的 Y 地址空间指针 W 寄存器（YWM）位于 MODCON<7:4> 中。当 YWM 被设置为除 15 之外的任何值且 YMODEN 位（MODCON<14>）置 1 时，Y 数据空间的模寻址被使能。

图 **4-1**： 模寻址操作示例


字节
地址
0x1100
0x1163
MOV #0x1100,W0
MOV W0, XMODSRT ; 设定模起始地址
MOV #0x1163,W0
MOV W0,XMODEND ; 设定模结束地址
MOV #0x8001,W0
MOV W0,MODCON ; 使能 W1、 X AGU 用于模寻址
MOV #0x0000,W0 ;W0 保持缓冲区填充值
MOV #0x1110,W1 ;W1 指向缓冲区
DO AGAIN,#0x31 ; 填充 50 个缓冲区单元
MOV W0, [W1++] ; 填充下一个单元
AGAIN: INC W0,W0 ; 填充值递增 1
起始地址 = 0x1100
结束地址 = 0x1163
长度 = 0x0032 个字

4.2.3 模寻址的应用

模寻址可以应用于与任何与 W 寄存器相关的有效地址（EA）计算中。重要的是要意识到，地址边界检查功能，不仅会检查地址是否正好在地址边界上，而且会检查地址是否小于或大于上限 （对于递增缓冲区）、是否低于下限 （对于递减缓冲区）。因此，地址变化可能会越过边界，但仍然可以正确调整。

注：	只有在使用执行前修改或执行后修改寻址模式来计算有效地址时，模修正有效地址才被写回寄存器。如果使用了地址偏移量（例如，[W7 + W2]），会进行地址修正，但寄存器的内容保持不变。

4.3 位反转寻址

位反转寻址用来简化基-2 FFT 算法的数据重新排序。位反转寻址为 X WAGU 所支持，仅限于数据写入。

地址修改量，可以是常数或寄存器的内容，可视为将其位顺序反转。地址源和目的仍然是正常的顺序。 于是，唯一需要反转的操作数就是地址修改量。

4.3.1 位反转寻址的实现

位反转寻址的使能方式如下：

1. MODCON 寄存器中 BWM （W 寄存器选择）的值，是除 15 以外的任何值（ 不能使用位反转寻址访问堆栈），且
2. XBREV 寄存器中 BREN 位置 1，且
3. 使用的寻址模式是预递增或后递增的寄存器间接寻址模式。

如果位反转缓冲区的长度是 $M = 2^N$ 字节，则数据缓冲区起始地址的最后 N 位必须为零。

XB<14:0> 是位反转地址修改量或 “中心点”（pivot point），通常是一个常数。对于 FFT 计算，其值等于 FFT 数据缓冲区长度的一半。

注：	所有位反转 EA 的计算都使用字数据（每个 EA 的 LSb 始终为零）。为了产生兼容地址（字节），要相应地调整 XB 的值。

使能位反转寻址时，仅对预递增或后递增的寄存器间接寻址、且仅对字数据写入，才会进行位反转寻址。对于任何其他寻址模式或对于字节数据，不会进行位反转寻址，而是生成正常的地址。在进行位反转寻址时，W 地址指针的增量将始终加上地址修改量（XB），与寄存器间接寻址模式相关的偏移量将被忽略。此外，由于要求是字数据， EA 的 LSb 被忽略 （且始终被清零）。

注：	不应同时使能模寻址和位反转寻址。如果用户试图这么做的话，对于 X WAGU，位反转寻址将优先，X WAGU 模寻址将被禁止。然而，在 X RAGU 中，模寻址继续起作用。

如果通过置 1 BREN （XBREV<15>）位使能了位反转寻址，那么，在写 XBREV 寄存器之后，不应马上进行要使用被指定为位反转指针的 W 寄存器的间接读操作。

图 4-2： 位反转地址示例

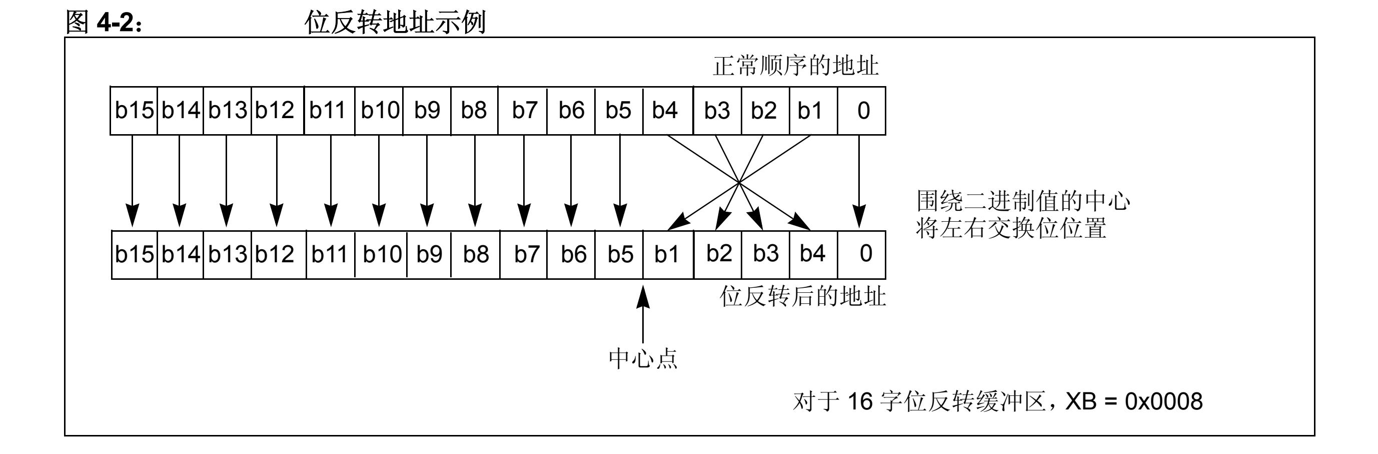


表 4-2： 位反转地址序列 （16 个项）

正常地址					位反转后的地址				
A3	A2	A1	A0	十进制	A3	A2	A1	A0	十进制
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	8
0	0	1	0	2	0	1	0	0	4
0	0	1	1	3	1	1	0	0	12
0	1	0	0	4	0	0	1	0	2
0	1	0	1	5	1	0	1	0	10
0	1	1	0	6	0	1	1	0	6
0	1	1	1	7	1	1	1	0	14
1	0	0	0	8	0	0	0	1	1
1	0	0	1	9	1	0	0	1	9
1	0	1	0	10	0	1	0	1	5
1	0	1	1	11	1	1	0	1	13
1	1	0	0	12	0	0	1	1	3
1	1	0	1	13	1	0	1	1	11
1	1	1	0	14	0	1	1	1	7
1	1	1	1	15	1	1	1	1	15

表 4-3： XBREV 寄存器的位反转地址修改量

缓冲区大小 （字）	XB<14:0> 位反转地址修改量 [1]
32768	0x4000
16384	0x2000
8192	0x1000
4096	0x0800
2048	0x0400
1024	0x0200
512	0x0100
256	0x0080
128	0x0040
64	0x0020
32	0x0010
16	0x0008
8	0x0004
4	0x0002
2	0x0001

注 1： 大于 256 字的修改量将超出 dsPIC30F2010 器件的数据存储空间范围。

5.0 中断

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

dsPIC30F2010 具有 24 个中断源和 4 个处理器异常（陷阱）。所有中断和异常的处理必须依据优先级机制进行仲裁。

CPU 负责读取中断向量表（IVT），并把中断向量中的地址传送给程序计数器。中断向量从程序数据总线向程序计数器的传送，是经由程序计数器输入侧 24 位宽的多路开关来进行的。

中断向量表（IVT）和备用中断向量表（AIVT）的位置靠近程序存储器的开始（0x000004）。IVT 和 AIVT 如图 5-1 所示。

在中断和处理器异常提交给处理器内核之前，中断控制器负责它们的预处理。使用下列集中式的特殊功能寄存器，对外设中断和陷阱进行允许、优先级分配和控制：

- IFS0<15:0>、IFS1<15:0> 和 IFS2<15:0>
  这三个寄存器中保存所有的中断请求标志。中断请求标志由其对应的外设或外部信号置 1，标志的清除则通过软件完成。
- IEC0<15:0>、IEC1<15:0> 和 IEC2<15:0>
  这三个寄存器保存所有的中断允许控制位。这些控制位用来单独允许来自外设或外部信号的中断。
- IPC0<15:0>... IPC11<10:0>
  与这些中断中的每一个中断有关的用户可分配中断优先级集中存放在这 12 个寄存器中。
- IPL<3:0>
  当前 CPU 优先级显式地存放在 IPL 位中。IPL<3>在CORCON寄存器中，而IPL<2:0> 则在处理器内核的状态寄存器（SR）中。
- INTCON1<15:0> 和 INTCON2<15:0>
  全局中断控制功能由这两个寄存器提供。INTCON1 包含处理器异常的控制和状态标志。INTCON2 寄存器控制外部中断请求信号的操作以及备用中断向量表的使用。

> 注：一旦产生中断条件，就将置 1 中断标志位，无论其对应的中断允许位状态为何。在允许一个中断之前，用户软件应确保相应的中断标志位被清零。

用户可以通过 IPCx 寄存器，为所有中断源分配 7 个优先级中的一个，优先级从 1 到 7。每个中断源都与一个中断向量关联，如所图 5-1 所示。优先级 7 和 1 分别代表最高和最低的可屏蔽优先级。

> 注：中断源优先级指定为 `0`，等同于禁止该中断。

如果 NSTDIS 位（INTCON1<15>）置 1，则禁止中断嵌套。因此，正在处理一个中断时，禁止处理新的中断，即使新中断的优先级比当前正在处理的中断优先级高。

> 注：一旦 NSTDIS 位置为 `1`，IPL 就变为只读。

某些中断具有专门的控制位，用于控制诸如边沿或电平触发中断、电平变化中断这样的功能。这些功能的控制仍然由产生中断的外设模块负责。

对于某些指令，在其执行期间如果 DISI 位（INTCON2<14>）保持置 1 的话，那么可以用 `DISI` 指令来禁止优先级为 6 或更低的中断的处理。

当中断得到服务时，PC 中装入存放在程序存储器中相应中断向量存储单元中的地址。IVT 中有 63 个不同的向量（见表 5-1）。这些向量存放在程序存储器从 0x000004 到 0x0000FE 的存储单元中（见图 5-1）。这些存储单元中存有 24 位地址；为了保持鲁棒性，如果在正常执行期间，PC 试图取任何这些字的话，就会产生地址错误陷阱。这避免了执行随机数据；PC 递减时意外进入向量空间，意外地把数据空间地址映射到了向量空间，或者 PC 到达可用程序空间的末尾继而返回到 0x000000，都将执行随机数据。执行 `GOTO` 指令跳转到向量空间，也将产生地址错误陷阱。

5.1 中断优先级

对于每个中断源，用户可分配的中断优先级位（IP<2:0>），位于 IPCx 寄存器中每个半字节的最低 3 位中。每个半字节的 bit 3 未使用，读为 0。这些位定义了用户分配给特定中断的优先级。

注：	用户可选择的优先级为从 0 到 7，0 是最低优先级， 7 是最高优先级。

由于特定的用户指定优先级会被分配给多个中断请求源，所以提供了一种在给定的用户分配优先级内分配优先级的方法。这个方法称为“自然顺序优先级”而且是决定性的。

自然顺序优先级由中断在向量表中的位置决定；只是在同时有多个中断——它们都具有相同的用户分配优先级——等待处理时，自然顺序优先级才会影响到中断操作。

表 5-1 列出了 dsPIC DSC 器件的中断编号和中断源，以及关联的向量编号。

注	1：	自然顺序优先级机制以 0 为最高优先级，以 53 为最低优先级。
	2：	自然顺序优先级编号与 INT 编号相同。

用户能够为每个中断分配七个优先级中的一个，这就意味着，对于一个具有低自然顺序优先级的中断，用户能够分配一个非常高的总优先级。例如，可以将低压检测（Low-Voltage Detect， PLVD ）的优先级分配为 7。INT0 （外部中断 0）可以分配为优先级 1，这样，它的有效优先级将非常低。

表 5-1：　dsPIC30F2010 中断向量表

中断编号	向量编号	中断源
最高自然顺序优先级		
0	8	INT0 – 外部中断 0
1	9	IC1 – 输入中断 1
2	10	OC1 – 输出比较 1
3	11	T1 – Timer 1
4	12	IC2 – 输入捕捉 2
5	13	OC2 – 输出比较 2
6	14	T2 – Timer 2
7	15	T3 – Timer 3
8	16	SPI1
9	17	U1RX – UART1 接收器
10	18	U1TX – UART1 发送器
11	19	ADC – ADC 转换结束
12	20	NVM – NVM 写结束
13	21	SI2C – I^2C 从中断
14	22	MI2C – I^2C 主中断
15	23	输入电平变化中断
16	24	INT1 – 外部中断 1
17	25	IC7 – 输入捕捉 7
18	26	IC8 – 输入捕捉 8
19	27	保留
20	28	保留
21	29	保留
22	30	保留
23	31	INT2 – 外部中断 2
24	32	保留
25	33	保留
26	34	保留
27	35	保留
28	36	保留
29	37	保留
30	38	保留
31	39	保留
32	40	保留
33	41	保留
34	42	保留
35	43	保留
36	44	INT3 – 外部中断 3
37	45	保留
38	46	保留
39	47	PWM – PWM 周期匹配
40	48	QEI – QEI 中断
41	49	保留
42	50	保留
43	51	FLTA – PWM 故障 A
44	52	保留
45-53	53-61	保留
最低自然顺序优先级		

5.2 复位过程

复位不是真正的异常，因为复位过程中并不涉及中断控制器。在响应强制清零 PC 的复位时，器件初始化其寄存器。然后，处理器开始在 0x000000 地址处执行程序。一条 GOTO 指令存放在第一个程序存储单元中，紧跟着的是 GOTO 指令的地址目标。处理器执行 GOTO，跳转到指定地址，然后开始指定目标（起始）地址处的操作。

5.2.1 复位的来源

除了外部复位和上电复位（POR）之外，还有 6 个错误条件会导致处理器复位。

- 看门狗定时器超时：
  看门狗定时器已经超时，表明处理器不再执行正确的代码流。
- 未初始化的 W 寄存器陷阱：
  试图把未初始化的 W 寄存器用作地址指针将导致复位。
- 非法指令陷阱：
  试图执行任何未使用的操作码，将产生非法指令陷阱。注意，如果由于指令流改变而在非法指令执行之前就被从流水线中舍弃的话，取非法指令并不产生非法指令陷阱。
- 欠压复位 （BOR）：
  检测到可能导致器件故障的电源电压短暂降低。
- 陷阱锁定：
  多个并发的陷阱条件将导致复位。

5.3 陷阱

可以将陷阱看作不可屏蔽中断，表明出现软件或硬件错误，陷阱遵循预定义的优先级，如表 5-1 所示。陷阱旨在为用户提供一种方法，改正在调试和在应用中工作时的错误操作。

> **注：** 如果用户不想在出现陷阱错误条件时采取纠正措施，那么必须在这些陷阱向量中装入缺省陷阱处理程序的地址，缺省陷阱处理程序仅包含 RESET 指令。否则，如果调用了包含非法地址的陷阱向量，将产生地址错误陷阱。

注意，很多陷阱条件只有在发生的时候才能检测到。因此，在陷阱异常处理之前允许有问题的指令完成执行。如果用户选择从错误中恢复，可能不得不改正导致陷阱的错误操作的结果。

陷阱有 8 个固定的优先级：优先级 8 到优先级 15，这意味着在陷阱处理期间， IPL3 始终置 1。

如果用户目前不在执行陷阱处理，而将 IPL<3:0> 设置为 0111（优先级 7），这样就禁止了所有中断，但仍然能够处理陷阱。

5.3.1 陷阱的来源

下列陷阱的优先级依次递增。然而，由于所有陷阱都可以嵌套，优先级的作用很小：

数学错误陷阱：

在下列四种情况下，产生数学错误陷阱：

1. 如果试图进行以零作除数的除法操作，除法操作将在周期边界处中止，并产生陷阱。
2. 如果使能的话，当对累加器 A 或 B 进行的算术操作导致了 bit 31 溢出，并且没有使用累加器警戒位时，将产生数学错误陷阱。
3. 如果使能的话，当对累加器 A 或 B 进行的算术操作导致了 bit 39 灾难性溢出，并且所有饱和被禁止，将产生数学错误陷阱。
4. 如果在移位指令中，指定的移位位数大于允许的最大移位位数，将产生陷阱。

地址错误陷阱：

当发生任何下列情形时，将产生陷阱：

1. 试图访问不对齐的数据字。
2. 试图从未实现的数据存储单元取数据。
3. 试图访问未实现的程序存储单元。
4. 试图从向量空间取指令。

注：	在 MAC 类指令中，数据空间被分成 X 数据空间 和 Y 数据 空间，未实现的 X 空间包括所有 Y 空间，而未实现的 Y 空间包括所有 X 空间。

5. 执行 `BRA #literal` 指令或 `GOTO #literal` 指令，其中 `literal` 是未实现的程序存储器地址。
6. 修改 PC 使其指向未实现的程序存储器地址后，执行指令。通过将值装入堆栈并执行 RETURN 指令可以修改 PC。

堆栈错误陷阱：

在下列条件下，产生堆栈错误陷阱：

1. 在堆栈指针中装入了一个大于堆栈限制的值（堆栈溢出）；堆栈限制值用户可编程，存放在 SPLIM 寄存器中。
2. 堆栈指针装入了一个小于 0x0800 的值 （简单堆栈下溢）。

振荡器故障陷阱：

如果外部振荡器出现故障，器件使用备用的内部 RC 振荡器工作，就会产生振荡器故障陷阱。

5.3.2 硬陷阱和软陷阱

在同一周期内可能会产生多个陷阱（例如，把不对齐的字写入堆栈中的溢出地址）。这时，图 5-1 所示的固定优先级就会起作用；为了完全纠正错误，这可能需要用户检查是否有其他等待处理的陷阱。

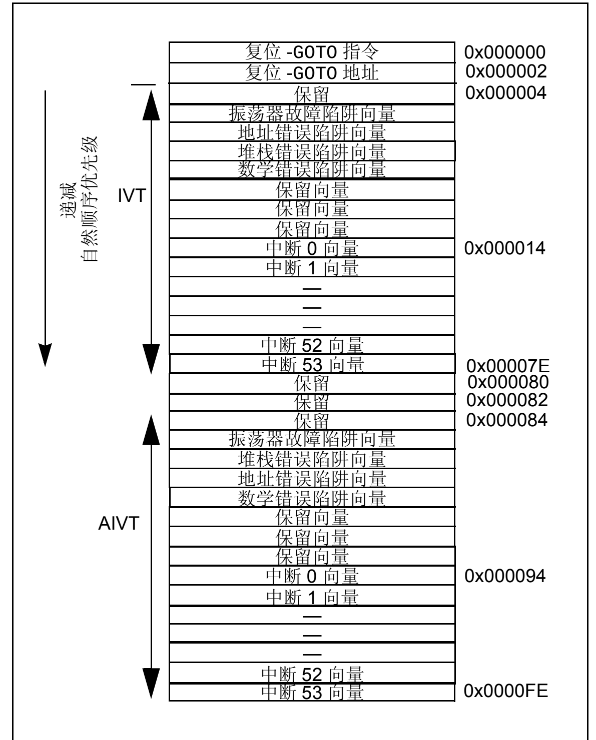
“软”陷阱包括优先级 8 到 11 的异常。算术错误陷阱（优先级 11）就属于这一类陷阱。

“硬”陷阱包括优先级 12 至 15 的异常。地址错误（优先级 12）、堆栈错误 （优先级 13）和振荡器错误 （优先级 14）陷阱就属于这一类。

每个硬陷阱产生时，在执行任何代码之前，必须先对它进行响应。在优先级较高的陷阱正在等待处理、被响应或正在处理过程中时，如果产生了较低优先级的陷阱，就会产生硬陷阱冲突。

器件在硬陷阱冲突时自动复位。复位发生时，把 TRAPR 状态位（RCON<15>）置 1，从而在软件中可以检测到这一条件。

图 5-1： 陷阱向量



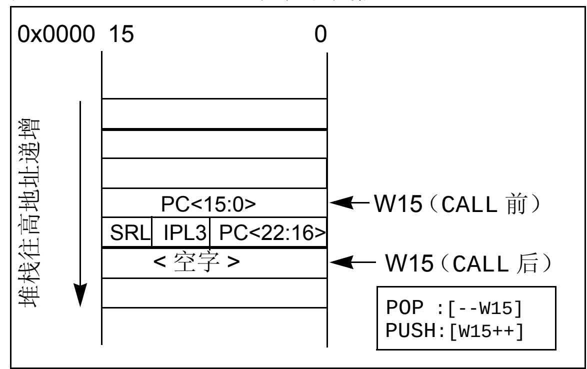
5.4 中断过程

在每个指令周期的开始， IFSx 寄存器都会采样全部的中断事件标志。IFSx 寄存器中标志位等于 1，表明有等待处理的中断请求（IRQ）。如果中断允许（IECx）寄存器中相应的位置 1，IRQ 将会导致中断产生。在 IRQ 采样后余下的指令周期中，将评估所有待处理中断请求的优先级。

如果有待处理的 IRQ，它的优先级比 IPL 位中的当前处理器优先级高，则处理器将被中断。

随即，处理器将把当前程序计数器和处理器状态寄存器的低字节（SRL）压入堆栈，如图 5-2 所示。状态寄存器低字节包含中断周期开始之前的处理器优先级。然后，处理器把中断的优先级装入状态寄存器。这将禁止所有低优先级的中断，直到中断服务程序（Interrupt Service Routine， ISR）结束为止。

图 5-2： 中断堆栈帧



> **注 1：** 用户能始终通过向 SR 写入一个新值以降低优先级。必须在降低处理器中断优先级之前，在中断服务程序中清除 IFSx 寄存器中的中断标志位以避免重复中断。
>
> **2：** 当处理中断时，IPL3 位（CORCON<3>）总是被清零。只有在进行陷阱处理时，该位才将会被置 1。

`RETFIE`（从中断返回）指令将把程序计数器和状态寄存器内容弹出堆栈，将处理器恢复到中断处理之前的状态。

5.5 备用向量表

在程序存储器中，中断向量表（IVT）之后是备用中断向量表（AIVT），如图 5-1 所示。INTCON2 寄存器中的 ALTIVT 位控制对 AIVT 的访问。如果 ALTIVT 位置 1，所有的中断和异常处理都将使用备用向量，而不是缺省向量。备用向量与缺省向量的组织方式是相同的。AIVT提供了一种在应用和支持环境之间切换的方法，从而支持仿真和调试功能，而不需要对中断向量进行再编程。这个特性也支持运行时在不同应用之间切换，以便评估各种软件算法。

如果不需要 AIVT，分配给 AIVT 的程序存储空间可以作其他用途。 AIVT 不是受保护的区域，用户可以对其自由编程。

5.6 快速现场保护

可以使用影子寄存器来保存现场。为 SR 中的 DC、N、OV、Z 和 C 位，以及寄存器 W0 到 W3 提供了影子寄存器。影子寄存器深度仅为一级。只能使用 `PUSH.S` 和 `POP.S` 指令来访问影子寄存器。

当处理器转移到中断向量开始处理中断时，可以使用 `PUSH.S` 指令，把上述寄存器的当前值保存到它们对应的影子寄存器中去。

如果某个优先级的 ISR 使用 `PUSH.S` 和 `POP.S` 指令来快速保存现场，那么优先级更高的 ISR 就不应再使用这两条指令。如果优先级较高的 ISR 使用了快速现场保护，那么在优先级较低的中断处理期间，用户必须保存关键寄存器。

5.7 外部中断请求

中断控制器支持多达 5 个外部中断请求信号， INT0-INT4。这些输入是边沿敏感的；它们需要从低至高或从高至低的跳变来产生中断请求。 INTCON2 寄存器的 3 个位 INT0EP-INT2EP，用来选择边沿检测电路的极性。

5.8 从休眠和空闲中唤醒

如果产生中断时器件处于休眠或空闲模式，中断控制器可以用来把处理器从休眠或空闲模式中唤醒。

如果中断控制器收到已允许的中断请求，且中断请求优先级足够高的话，则标准中断请求将提交给处理器。同时，处理器将从休眠或空闲中唤醒，开始执行处理中断请求所需的中断服务程序。

表 **5-2**： 中断控制寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
INTCON1	0080	NSTDIS	—	—	—	—	OVATE	OVBTE	COVTE	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000 0000 0000 0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000 0000 0000 0000
IFS0	0084	CNIF	MI2CIF	SI2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SPI1IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0IF	0000 0000 0000 0000
IFS1	0086	—	—	—	—	—	—	—	—	INT2IF	—	—	—	—	IC8IF	IC7IF	INT1IF	0000 0000 0000 0000
IFS2	0088	—	—	—	—	FLTAIF	—	—	QEIIF	PWMIF	—	—	—	—	—	—	—	0000 0000 0000 0000
IEC0	008C	CNIE	MI2CIE	SI2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SPI1IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000
IEC1	008E	—	—	—	—	—	—	—	—	INT2IE	—	—	—	—	IC8IE	IC7IE	INT1IE	0000 0000 0000 0000
IEC2	0090	—	—	—	—	FLTAIE	—	—	QEIIE	PWMIE	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC0	0094	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			0100 0100 0100 0100
IPC1	0096	—	T31P<2:0>			—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			0100 0100 0100 0100
IPC2	0098	—	ADIP<2:0>			—	U1TXIP<2:0>			—	U1RXIP<2:0>			—	SPI1IP<2:0>			0100 0100 0100 0100
IPC3	009A	—	CNIP<2:0>			—	MI2CIP<2:0>			—	SI2CIP<2:0>			—	NVMIP<2:0>			0100 0100 0100 0100
IPC4	009C	—	—	—	—	—	IC8IP<2:0>			—	IC7IP<2:0>			—	INT1IP<2:0>			0100 0100 0100 0100
IPC5	009E	—	INT2IP<2:0>			—	—	—	—	—	—	—	—	—	—	—	—	0100 0000 0000 0000
IPC6	00A0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC7	00A2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC8	00A4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC9	00A6	—	PWMIP<2:0>			—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC10	00A8	—	FLTAIP<2:0>			—	—	—	—	—	—	—	—	—	QEIIP<2:0>			0100 0000 0000 0100
IPC11	00AA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

6.0 闪存程序存储器

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

dsPIC30F 系列器件包含用于执行用户代码的闪存程序存储器。用户可以使用以下两种方法对闪存程序存储器编程：

1. 在线串行编程 （ICSP™）功能
2. 运行时自编程 （RTSP）

6.1 在线串行编程 （ICSP）

可以在最终应用电路中对 dsPIC30F 器件进行串行编程。实现很简单，仅需一根编程时钟线 （称为 PGC）、一根编程数据线 （称为 PGD），以及电源线 （VDD）、接地线 （VSS）和主复位线 （$\overline{\text{MCLR}}$）。这使得用户可以使用未编程的器件来生产电路板，在产品发货前才对数字信号控制器进行编程。这样做还可将最新固件或用户定制的固件编程到器件中。

6.2 运行时自编程 （RTSP）

使用 TBLRD （表读）和 TBLWT （表写）指令来实现 RTSP。

使用 RTSP，用户可以一次擦除 32 个指令字（96 字节）的程序存储单元，一次可写入 32 个指令字 （96 字节）的程序存储数据。

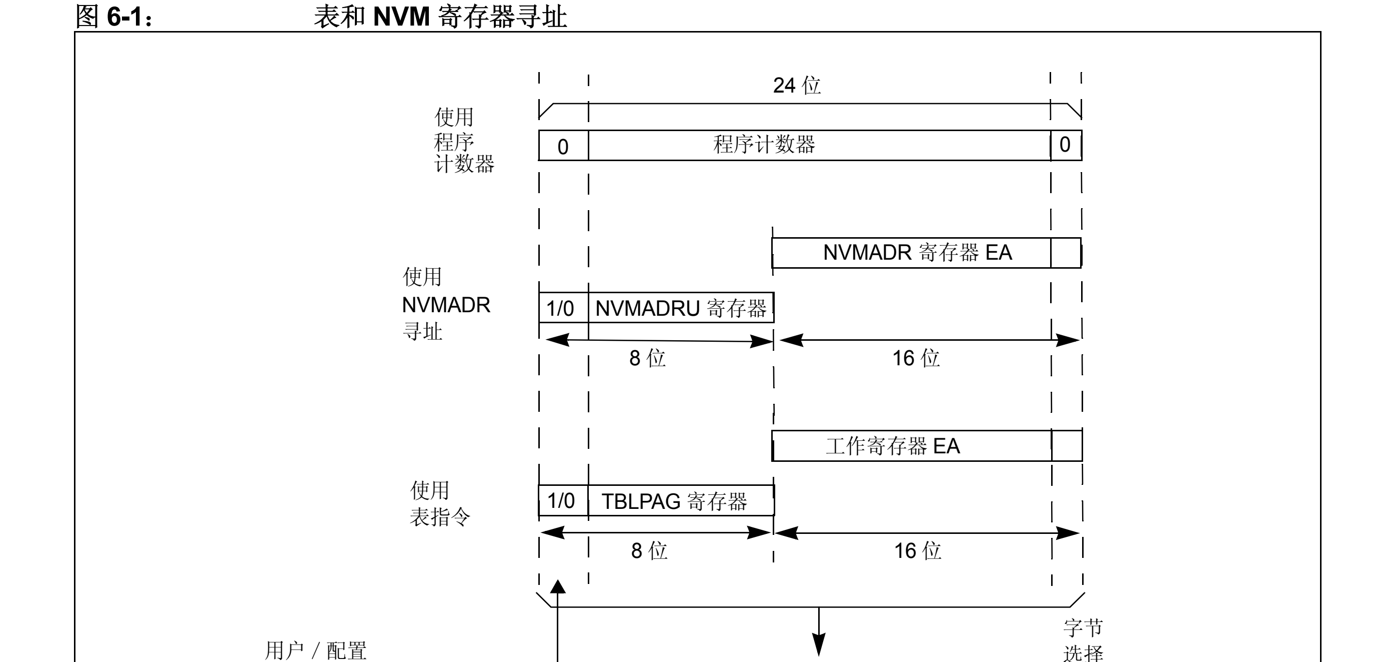
6.3 表指令操作概要

TBLRDL 和 TBLWTL 指令用来读或写程序存储器的 bit <15:0>。 TBLRDL 和 TBLWTL 可以以字或字节模式访问程序存储器。

TBLRDH 和 TBLWTH 指令用来读或写程序存储器的 bit <23:16>。TBLRDH 和 TBLWTH 可以以字或字节模式访问程序存储器。

使用 TBLPAG 寄存器的 bit <7:0> 和来自表指令中指定的 W 寄存器的 EA，形成 24 位的程序存储器地址，如图 6-1 所示。

图 6-1： 表和 NVM 寄存器寻址



6.4 RTSP 操作

dsPIC30F 闪存程序存储器组织成行和板（panel）。每行由 32 个指令字（96 字节）组成。每个板由 128 行组成（4K × 24 个指令字）。RTSP 允许用户每次擦除一行（32 个指令字）、每次编程 32 条指令。RTSP 可以用来对多个程序存储器板进行编程，但在每个板的边界处必须更改表指针。

程序存储器的每个板都包含写锁存器，它能够保存 32 个指令字的编程数据。在实际编程操作前，等待写入的数据必须先装入板的写锁存器。要写入板的数据按顺序装载到写锁存器中：指令 0，指令 1，依此类推。装载的指令字必须始终来自 32 个指令字的地址边界。

RTSP 编程的基本步骤是先建立一个表指针，然后执行一系列 TBLWT 指令，装载写锁存器。通过把 NVMCON 寄存器中的特殊位置 1，进行编程。装载 32 条指令需要 32 条 TBLWTL 和 4 条 TBLWTH 指令。如果需要对多个板进行编程，应该为下一组要写入的多个写锁存器修改表指针。

全部表写操作都是单字写入（2 个指令周期），因为只需要写表锁存器。编程每一行需要一个编程周期。

在整个 VDD 范围内，正常操作期间，闪存程序存储器是可读、可写且可擦除的。

6.5 控制寄存器

四个特殊功能寄存器用来读、写闪存程序存储器，它们是：

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

6.5.1 NVMCON 寄存器

NVMCON 寄存器控制要擦除的存储块、要编程的存储器类型以及编程周期的启动。

6.5.2 NVMADR 寄存器

NVMADR 寄存器用来存放有效地址的两个低字节。它捕捉已执行的上一条表指令的 EA<15:0>，选择要写的行。

6.5.3 NVMADRU 寄存器

NVMADRU 寄存器用来存放有效地址的高字节。它捕捉已执行的上一条表指令的 EA<23:16>。

6.5.4 NVMKEY 寄存器

NVMKEY 是只写寄存器，用于写保护。要启动编程或擦除过程，用户必须把 0x55 和 0xAA 连续写入 NVMKEY 寄存器。更多细节，请参阅**第 6.6 节 “编程操作”**。

注：	用户也可直接写 NVMADR 和 NVMADRU 寄存器，指定擦除或编程的程序存储器地址。

6.6 编程操作

在 RTSP 模式下，对内部闪存进行编程或擦除需要完整的编程过程。编程操作持续时间的标称值为 2 ms，编程操作结束之前，处理器将停顿（等待）。把 WR 位（NVMCON<15>）置 1 开始编程操作，操作结束后将自动清零 WR 位。

6.6.1 闪存程序存储器的编程算法

用户一次可以擦除、编程闪存程序存储器的一行。一般过程如下：

1. 读一行闪存程序存储器（32 个指令字），把读出的数据作为数据 “镜像”保存到数据 RAM。
2. 用期望的新数据更新数据镜像。
3. 擦除闪存程序存储器行。
   a) 设置 NVMCON 寄存器以允许对闪存程序存储器多字擦除，并置 1 WREN 位。
   b) 把要擦除的行地址写入寄存器 NVMADRU/NVMADR。
   c) 把 “55”写入 NVMKEY。
   d) 把 “AA”写入 NVMKEY。
   e) 置 1 WR 位。这将开始擦除周期。
   f) 在擦除周期中 CPU 将停顿。
   g) 当擦除周期结束时 WR 位会被清零。
4. 从数据 RAM“镜像”中，把 32 个指令字的数据写入闪存程序存储器写锁存器。
5. 将 32 个指令字写入闪存程序存储器。
   a) 设置 NVMCON 寄存器以允许对闪存程序存储器多字编程，并置 1 WREN 位。
   b) 把 “55”写入 NVMKEY。
   c) 把 “AA”写入 NVMKEY。
   d) 置 1 WR 位。这将开始编程周期。
   e) 在编程周期中 CPU 将停顿。
   f) 当编程周期结束时 WR 位会被硬件清零。
6. 根据需要，重复步骤1 到5，对所需的闪存程序存储器空间进行编程。

6.6.2 擦除程序存储器的一行

例 6-1 所示的代码序列，可以用来擦除一行程序存储器（32 个指令字）。

例 6-1： 擦除程序存储器的一行

```
; 设置 NVMCON 进行擦除操作和多字写
; 选择程序存储器并使能写操作
        MOV     #0x4041,W0                  ;
        MOV     W0,NVMCON                   ; 初始化 NVMCON SFR
; 初始化指针以指向要被擦除的行
        MOV     #tblpage(PROG_ADDR),W0      ;
        MOV     W0,NVMADRU                  ; 初始化 PM 页边界 SFR
        MOV     #tbloffset(PROG_ADDR),W0    ; 初始化页内 EA[15:0] 指针
        MOV     W0, NVMADR                  ; 初始化 NVMADR SFR
        DISI    #5                          ; 在以后的 5 个指令周期内阻止所有优先级 <7 的中断
        MOV     #0x55,W0
        MOV     W0,NVMKEY                   ; 写 0x55 键
        MOV     #0xAA,W1                    ;
        MOV     W1,NVMKEY                   ; 写 0xAA 键
        BSET    NVMCON,#WR                  ; 开始擦除过程
        NOP                                 ; 在擦除之后插入两条 NOP 指令
        NOP                                 ; 命令结束
```

6.6.3 装载写锁存器

例 6-2 所示的指令序列，可以用来装载 96 字节的写锁存器。装载由表指针选择的写锁存器，需要 32 条 TBLWTL 指令和 32 条 TBLWTH 指令。

例 6-2： 装载写锁存器

```
; 设置一个指针指向要写入的第一个程序存储单元
; 选择程序存储器并使能写操作
        MOV     #0x0000,W0                      ;
        MOV     W0,TBLPAG                       ; 初始化 PM 页边界 SFR
        MOV     #0x6000,W0                      ; 程序存储器地址示例
; 执行 TBLWT 指令以写入锁存器
; 0th_program_word
        MOV     #LOW_WORD_0,W2                  ;
        MOV     #HIGH_BYTE_0,W3                 ;
        TBLWTL  W2,[W0]                         ;将 PM 低位字内容写入程序锁存器
        TBLWTH  W3,[W0++]                       ;将 PM 高字节内容写入程序锁存器
; 1st_program_word
        MOV     #LOW_WORD_1,W2                  ;
        MOV     #HIGH_BYTE_1,W3                 ;
        TBLWTL  W2,[W0]                         ; 将 PM 低位字内容写入程序锁存器
        TBLWTH  W3,[W0++]                       ; 将 PM 高字节内容写入程序锁存器
;  2nd_program_word
        MOV     #LOW_WORD_2,W2                  ;
        MOV     #HIGH_BYTE_2,W3                 ;
        TBLWTL  W2, [W0]                        ; 将 PM 低位字内容写入程序锁存器
        TBLWTH  W3, [W0++]                      ; 将 PM 高字节内容写入程序锁存器
        •
        •
        •
; 31st_program_word
        MOV     #LOW_WORD_31,W2                 ;
        MOV     #HIGH_BYTE_31,W3                ;
        TBLWTL  W2, [W0]                        ; 将 PM 低位字内容写入程序锁存器
        TBLWTH  W3, [W0++]                      ; 将 PM 高字节内容写入程序锁存器
```

注：在例 6-2 中， W3 高字节的内容将不起作用。

6.6.4 启动编程过程

出于保护的目的，必须使用 NVMKEY 的写启动序列，以便允许进行任何擦除或编程操作。在编程命令执行后，用户必须等待一段时间（编程时间），直至编程完成。编程过程开始之后，紧接着的应该是两条 NOP 指令。

例 6-3： 启动编程过程

```
        DISI    #5                      ; 在以后的 5 个指令周期内阻止所有优先级 <7 的中断
        MOV     #0x55,W0
        MOV     W0,NVMKEY               ; 写入 0x55 键
        MOV     #0xAA,W1                ;
        MOV     W1,NVMKEY               ; 写入 0xAA 键
        BSET    NVMCON,#WR              ; 开始擦除过程
        NOP                             ; 在擦除之后插入两条 NOP 指令
        NOP                             ; 命令结束
```

表 6-1： NVM 寄存器映射

文件寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位时状态
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	TWRI	—	PROGOP<6:0>							0000 0000 0000 0000
NVMADR	0762	NVMADR<15:0>																uuuu uuuu uuuu uuuu
NVMADRU	0764	—	—	—	—	—	—	—	—	NVMADR<23:16>								0000 0000 uuuu uuuu
NVMKEY	0766	—	—	—	—	—	—	—	—	KEY<7:0>								0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

7.0 数据 EEPROM 存储器

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

在整个 VDD 范围内的正常操作期间，数据 EEPROM 存储器是可读且可写的。数据 EEPROM 存储器直接映射到程序存储器地址空间。

用来读写闪存程序存储器的四个特殊功能寄存器，也用来访问数据 EEPROM 存储器。如在第 6.5 节中所述，这些寄存器是：

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

EEPROM 数据存储器允许读写单字和 16 字长的块。针对数据存储器时，NVMADR 与 NVMADRU 寄存器配合使用，用来寻址要访问的 EEPROM 地址单元。TBLRDL 和 TBLWTL 指令用来读写数据 EEPROM。dsPIC30F 器件有最多 1KB 的数据 EEPROM，地址范围从 0x7FFC00 到 0x7FFFFE。

字写操作之前，应该先擦除相应的存储单元。写操作完成通常需要 2 ms，不过它将随电压和温度的变化而变化。

对数据 EEPROM 进行编程或擦除操作，不会停止指令流。在启动另一次数据 EEPROM 写 / 擦除操作之前，用户需要等待一段适当的时间。在编程或擦除操作进行过程中，尝试读数据 EEPROM 将返回不确定的数据。

控制位 WR 启动写操作，这与 闪存程序存储器写类似。可通过软件置 1 WR 位，但不能对其清零。写操作完成时，硬件将清零 WR 位。不能用软件清零 WR 位，这样避免了意外或提早结束写操作。

WREN 位使能时将允许进行写操作。上电时，WREN 位清零。当写操作被正常工作期间的 $\overline{\text{MCLR}}$ 复位或 WDT 超时所中断时，WRERR 位置 1。在这样的情形下，复位后，用户可以检查 WRERR 位，重写存储单元。地址寄存器 NVMADR 保持不变。

> 注：在写操作完成时，IFS0 寄存器中的中断标志位 NVMIF 将置 1。NVMIF 位必须用软件清零。

7.1 读数据 EEPROM

TBLRD 指令读取位于当前程序字地址的字。下面的示例使用 W0 作为指向数据 EEPROM 的指针。结果存放在寄存器 W4 中，如例 7-1 所示。

例 7-1： 读数据 EEPROM

```
MOV     #LOW_ADDR_WORD,W0   ; 初始化指针
MOV     #HIGH_ADDR_WORD,W1
MOV     W1,TBLPAG
TBLRDL  [ W0 ], W4          ; 读数据 EEPROM
```

7.2 擦除数据 EEPROM

7.2.1 擦除数据 EEPROM 的一块

为了擦除数据 EEPROM 的一块，首先 NVMADRU 和 NVMADR 寄存器必须指向要擦除的存储块。将 NVMCON 配置为擦除数据 EEPROM 块，并置 1 NVMCON寄存器中的WR位和WREN位。置1 WR 位，启动擦除操作，如例 7-2 所示。

例 7-2： 擦除数据 EEPROM 的一块

```
; 选择要擦除的数据 EEPROM 块，使能 WREN 位
    MOV     #4045,W0
    MOV     W0,NVMCON                          ; 初始化 NVMCON SFR

; 在写入键序列后通过置 1 WR 位以启动擦除过程
    DISI    #5                                 ; 在以后的 5 个指令周期内阻止所有优先级 <7 的中断
    MOV     #0x55,W0                           ;
    MOV     W0,NVMKEY                          ; 写入 0x55 键
    MOV     #0xAA,W1                           ;
    MOV     W1,NVMKEY                          ; 写入 0xAA 键
    BSET    NVMCON,#WR                         ; 启动擦除过程
    NOP
    NOP
; 擦除过程将在 2mS 内完成。 CPU 在数据擦除周期中将不会处于停顿状态。
; 用户可查询 WR 位的状态，使用 NVMIF 或定时器 IRQ 来确定擦除操作是否完成。
```

7.2.2 擦除数据 EEPROM 的一个字

NVMADRU和NVMADR寄存器必须指向要擦除的存储块。在 NVMCON 寄存器中，选择擦除数据 EEPROM 的一个字，置1 WR 位和 WREN 位。置 1 WR 位，启动擦除操作，如例 7-3 所示。

例 7-3： 擦除数据 EEPROM 的一个字

```
; 选择要擦除的数据 EEPROM 字，使能 WREN 位
    MOV     #4044,W0
    MOV     W0,NVMCON

; 在写入键序列后通过置 1 WR 位以开始擦除周期。
    DISI    #5                                 ;在以下的 5 个指令周期内阻止所有优先级 <7 的中断
    MOV     #0x55,W0               ;
    MOV     W0,NVMKEY              ; 写入 0x55 键
    MOV     #0xAA,W1               ;
    MOV     W1,NVMKEY              ; 写入 0xAA 键
    BSET    NVMCON,#WR             ; 启动擦除过程
    NOP
    NOP
; 擦除过程将在 2mS 内完成。 PU 在数据擦除周期中将不会处于停顿状态。
; 用户可查询 WR 位的状态，使用 NVMIF 或定时器 IRQ 来确定擦除操作是否完成。
```


7.3 写数据 EEPROM

要写 EEPROM 数据存储单元，必须按照下列顺序进行：

1. 擦除数据 EEPROM 的字。
   a) 在 NVMCON 寄存器中，选择字擦除数据 EEPROM，并置 1 WREN 位。
   b) 把要擦除的字的地址写入 NVMADR/NVMADRU。
   c) 允许 NVM 中断 （可选）。
   d) 把 “55” 写入 NVMKEY。
   e) 把 “AA” 写入 NVMKEY。
   f) 置 1 WR 位。这将开始擦除周期。
   g) 查询 NVMIF 位或等待 NVMIF 中断。
   h) 当擦除周期结束时 WR 位会被清零。
2. 把数据字写入数据 EEPROM 的写锁存器中。
3. 把 1 个数据字编程到数据 EEPROM 中。
   a) 在 NVMCON 寄存器中，选择字编程数据 EEPROM，并置 1 WREN 位。
   b) 允许 NVM 写完成中断 （可选）。
   c) 把 “55” 写入 NVMKEY。
   d) 把 “AA” 写入 NVMKEY。
   e) 置 1 WR 位。这将开始编程周期。
   f) 查询 NVMIF 位或等待 NVM 中断。
   g) 当写周期结束时 WR 位会被清零。

对于每个字，如果上述序列（把 0x55 写入 NVMKEY，把 0xAA 写入 NVMCON，然后置 1 WR 位）没有严格遵守的话，写操作不会启动。强烈建议在这段代码执行期间，禁止中断。

此外，必须置 1 NVMCON 中的 WREN 位来使能写操作。这样的机制，防止由于不可预料的代码执行而意外地写数据 EEPROM。除非在更新 EEPROM 时，否则 WREN 位应该始终保持清零。硬件不能清零 WREN 位。

在写过程启动之后，清零 WREN 位将不会影响当前的写周期。禁止把 WR 位置 1，除非 WREN 位已置 1。必须在前一条指令中置 1 WREN 位。不能在同一条指令中置 1 WR 和 WREN 位。

写周期结束时，硬件清零 WR 位，非易失性存储器写完成中断标志位 （NVMIF）置 1。用户可以允许这个中断，或者查询这个标志位。 NVMIF 必须由软件清零。

7.3.1 写数据 EEPROM 的一个字

一旦用户擦除了要编程的字，则表写指令用来写写锁存器，如例 7-4 所示。

例 7-4： 数据 EEPROM 字写入

```
; 指向数据存储器
    MOV         #LOW_ADDR_WORD,W0              ; 初始化指针
    MOV         #HIGH_ADDR_WORD,W1
    MOV         W1,TBLPAG
    MOV         #LOW(WORD),W2                  ; 获得数据
    TBLWTL      W2,[ W0]                       ; 写入数据
; NVMADR 捕捉上一次表访问地址
; 选择要写的数据 EEPROM 字
    MOV         #0x4004,W0
    MOV         W0,NVMCON

; 写入键以允许写操作
    DISI   #5                        ; 在以下 5 个指令周期内阻止所有优先级 <7 的中断
    MOV         #0x55,W0
    MOV         W0,NVMKEY                      ; 写入 0x55 键
    MOV         #0xAA,W1
    MOV         W1,NVMKEY                      ; 写入 0xAA 键
    BSET        NVMCON,#WR                     ; 启动编程过程
    NOP
    NOP
; 写过程将在 2mS 内完成。 CPU 在数据写周期中不会处于停顿状态。
; 用户可查询 WR 位的状态，使用 NVMIF 或定时器 IRQ 来确定写操作是否完成。
```

7.3.2 写数据 EEPROM 的一个存储块

要写入数据 EEPROM 的一个存储块，应首先写入所有 16 个锁存器，然后设置 NVMCON 寄存器并对存储块进行编程。

例 7-5： 数据 EEPROM 块写入

```
    MOV       #LOW_ADDR_WORD,W0  ; 初始化指针
    MOV       #HIGH_ADDR_WORD,W1
    MOV       W1,TBLPAG
    MOV       #data1,W2          ; 获取第 1 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data2,W2          ; 获取第 2 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data3,W2          ; 获取第 3 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data4,W2          ; 获取第 4 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data5,W2          ; 获取第 5 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data6,W2          ; 获取第 6 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data7,W2          ; 获取第 7 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data8,W2          ; 获取第 8 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data9,W2          ; 获取第 9 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data10,W2         ; 获取第 10 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data11,W2         ; 获取第 11 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data12,W2         ; 获取第 12 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data13,W2         ; 获取第 13 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data14,W2         ; 获取第 14 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data15,W2         ; 获取第 15 个数据
    TBLWTL    W2,[ W0]++         ; 写入数据
    MOV       #data16,W2         ; 获取第 16 个数据
    TBLWTL    W2,[ W0]++         ; 写数据。 NVMADR 捕捉上一次表访问地址。
    MOV       #0x400A,W0         ; 选择要进行多字写的数据 EEPROM 块
    MOV       W0,NVMCON          ; 写入键以允许编程操作
    DISI      #5                 ; 在以后 5 个指令周期内阻止所有优先级 <7 的中断
    MOV       #0x55,W0
    MOV       W0,NVMKEY          ; 写入键 0x55
    MOV       #0xAA,W1
    MOV       W1,NVMKEY          ; 写入键 0xAA
    BSET      NVMCON,#WR         ; 启动写周期
    NOP
    NOP
```

7.4 写校验

根据不同的应用，良好的编程习惯可能要求把已写入存储器中的值与原始值进行校验。如果在应用中进行了量非常大的写操作，从而可能接近规格说明中的上限时，应该进行写校验。

7.5 防止误写入

某些情况下，并不打算写数据 EEPROM 存储器。器件内置了各种机制来防止误写入 EEPROM。上电时，WREN 位清零；而且，上电延时定时器禁止 EEPROM 写操作。

写启动序列与 WREN 位一起，有助于防止欠压、电源干扰或软件故障期间的意外写入。

8.0 I/O 端口

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

所有的器件引脚（除 VDD、VSS、MCLR 和 OSC1/CLKI 之外），均为外设与并行 I/O 端口所共用。

全部 I/O 输入端口都是施密特触发器输入，以便增强抗干扰性。

8.1 并行 I/O （PIO）端口

如果外设使能，并且外设正在使用其关联引脚时，引脚将不再作为通用 I/O 引脚使用。这时，可以读 I/O 引脚，但是对应并行端口位的输出驱动器将被禁止。如果外设使能，但外设不在使用某引脚，则该引脚可以被端口驱动。

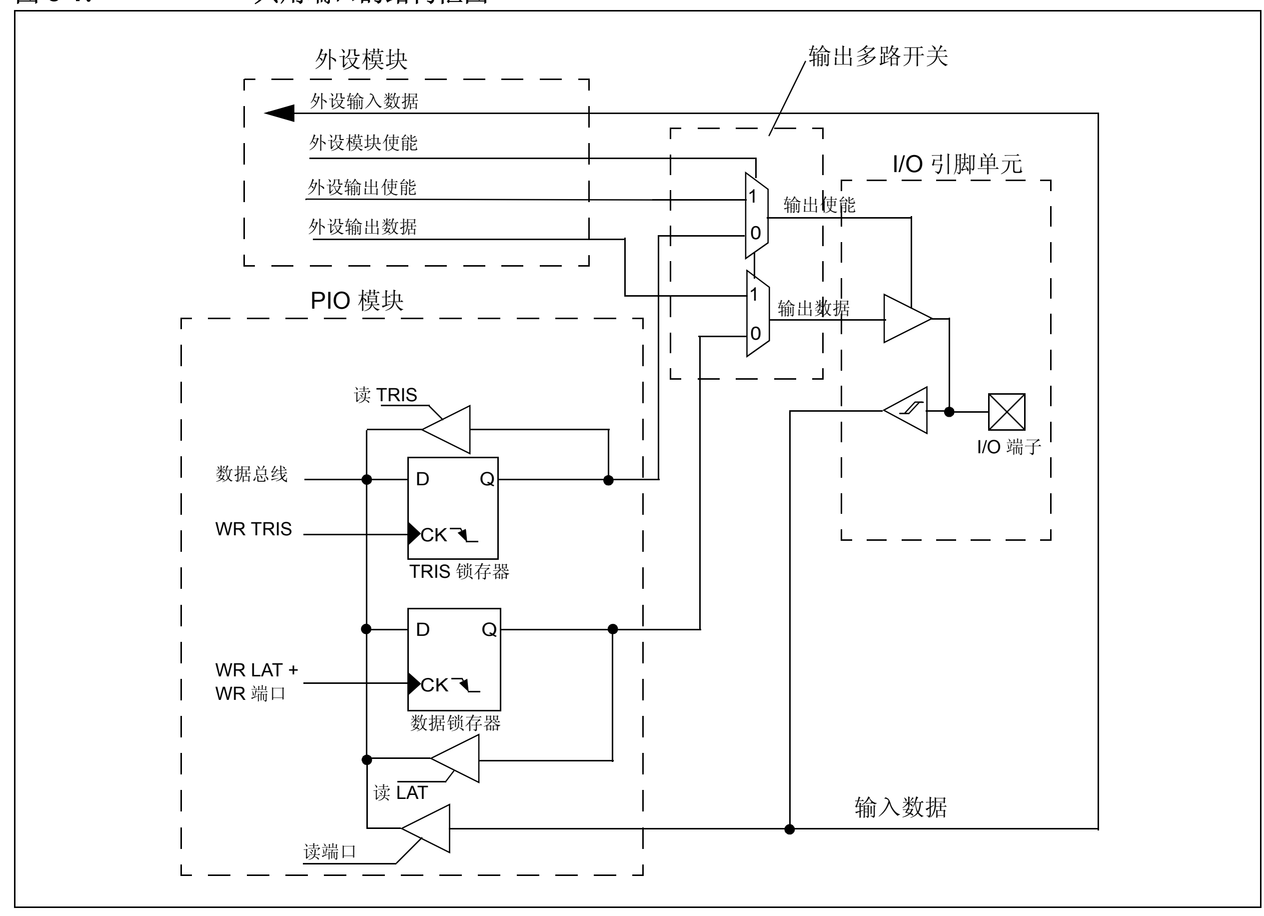
所有的端口引脚，都有三个与端口引脚的操作直接相关的寄存器。数据方向寄存器 （TRISx）决定引脚是输入还是输出。如果数据方向位是 1，则引脚是输入。复位后，全部端口引脚均定义为输入。读锁存器 （LATx），读的是锁存值；写锁存值，写的是锁存器（LATx）。读端口（PORTx），读的是端口引脚值，写端口引脚，写的是锁存器 （LATx）。

任何位及其关联的数据和控制寄存器，如果对于特定器件而言是无效的话，则将被禁止。这意味着，相应的 LATx 和 TRISx 寄存器以及端口引脚，都将读为零。

当端口引脚与另一个外设共用或与只定义为输入的功能共用时，它将被视为专用端口，因为没有任何其他竞争的输出资源。 INT4 引脚就是例子。

与外设共用引脚的并行 I/O （PIO）端口一般屈从于外设。外设的输出缓冲数据和控制信号提供给一对多路开关。多路开关选择外设还是相关的端口将拥有 I/O 引脚输出数据和控制信号的所有权。图 8-1 显示了端口是如何与其他外设共用，以及与外设连接的相关 I/O 引脚单元。表 8-1 给出了从 PORTB 到 PORTF 的共用端口的寄存器的格式。

图 8-1： 共用端口的结构框图



8.2 配置模拟端口引脚

使用 ADPCFG 和 TRIS 寄存器控制 A/D 端口引脚的操作。对于要用作模拟输入的端口引脚，必须将其相应的 TRIS 位置 1（输入）。如果 TRIS 位清零（输出），则将转换数字输出电平 （VOH 或 VOL）。

当读取端口寄存器时，所有配置为模拟输入通道的引脚，都将读为零 （低电平）。

配置为数字输入的引脚，将不会对模拟输入进行转换。当将模拟电压加在任何定义为数字输入的引脚 （包括 ANx 引脚）上时，可能导致输入缓冲器消耗的电流超出器件规范中规定的值。

8.2.1 I/O 端口读 / 写时序

在端口方向变化或端口写操作以及同一端口的读操作之间需要一个指令周期的等待时间。通常情况下插入一条 NOP 指令。

例 8-1： 端口读 / 写操作示例

```
MOV 0xFF00, W0  ; 将 PORTB<15:8> 配置为输入
MOV W0, TRISBB  ; 将 PORTB<7:0> 配置为输出
NOP             ; 延迟 1 个周期
btssPORTB, #13  ; 下一条指令
```

8.3 输入电平变化通知模块

输入电平变化通知模块，使得 dsPIC30F 器件能够产生中断请求并传递给处理器，以响应所选取输入引脚上的状态变化。即便是在休眠模式下、时钟被禁止时，该模块也能检测输入的状态变化。对于在输入状态变化时产生中断请求，可以选择 （使能）的外部信号多达 22 个（CN0 到 CN21）。

表 8-1： dsPIC30F2010 端口寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
TRISB	02C6	—	—	—	—	—	—	—	—	—	—	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	0000 0000 0011 1111
PORTB	02C8	—	—	—	—	—	—	—	—	—	—	RB5	RB4	RB3	RB2	RB1	RB0	0000 0000 0000 0000
LATB	02CA	—	—	—	—	—	—	—	—	—	—	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	0000 0000 0000 0000
TRISC	02CC	TRISC15	TRISC14	TRISC13	—	—	—	—	—	—	—	—	—	—	—	—	—	1110 0000 0000 0000
PORTC	02CE	RC15	RC14	RC13	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
LATC	02D0	LATC15	LATC14	LATC13	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
TRISD	02D2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TRISD1	TRISD0	0000 0000 0000 0111
PORTD	02D4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RD1	RD0	0000 0000 0000 0000
LATD	02D6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LATD1	LATD0	0000 0000 0000 0000
TRISE	02D8	—	—	—	—	—	—	—	TRISE8	—	—	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	0000 0001 0011 1111
PORTE	02DA	—	—	—	—	—	—	—	RE8	—	—	RE5	RE4	RE3	RE2	RE1	RE0	0000 0000 0000 0000
LATE	02DC	—	—	—	—	—	—	—	LATE8	—	—	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	0000 0000 0000 0000
TRISF	02DE	—	—	—	—	—	—	—	—	—	—	—	—	TRISF3	TRISF2	—	—	0000 0000 0000 1100
PORTF	02E0	—	—	—	—	—	—	—	—	—	—	—	—	RF3	RF2	—	—	0000 0000 0000 0000
LATF	02E2	—	—	—	—	—	—	—	—	—	—	—	—	LATF3	LATF2	—	—	0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

表 8-2： 输入电平变化通知寄存器映射 （BIT 15-0）

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
CNEN1	00C0	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000 0000 0000 0000
CNEN2	00C2	—	—	—	—	—	—	—	—	—	—	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000 0000 0000 0000
CNPU1	00C4	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000 0000 0000 0000
CNPU2	00C6	—	—	—	—	—	—	—	—	—	—	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

9.0 TIMER1 模块

> **注：**本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

本章介绍了 16 位通用 Timer1 模块以及相关的工作模式。图 9-1 给出了 16 位 Timer1 模块的简化框图。

> **注：** Timer1 是 A 类定时器。请参见本文挡**第 22.0 节“电气特性”**中 A 类定时器的规范。

下面各小节将详细说明定时器的工作模式，包括其设置、控制寄存器以及相应的框图。

Timer1模块是16位的定时器，可以用作实时时钟的时间计数器，或者用作自由运行的时段定时器 / 计数器。 16 位定时器有下列模式：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

而且，支持下列操作特征：

- 定时器门控操作
- 可选择的预分频比设置
- CPU 空闲和休眠模式下的定时器操作
- 在 16 位周期寄存器匹配时产生中断，或者在外部门控信号下降沿产生中断

通过设定 16 位 SFR T1CON 中的相应位来决定这些工作模式。图 9-1 所示为 16 位 Timer1 模块的框图。

16 位定时器模式：在 16 位定时器模式下，定时器将在每个指令周期递增，直到等于 PR1 中预先装入的值，然后复位至 0，继续计数。

当 CPU 进入空闲模式时，定时器将停止递增，除非 TSIDL（T1CON<13>）位 = 0。如果 TSIDL = 1，则定时器模块逻辑将在 CPU 空闲模式结束后继续递增。

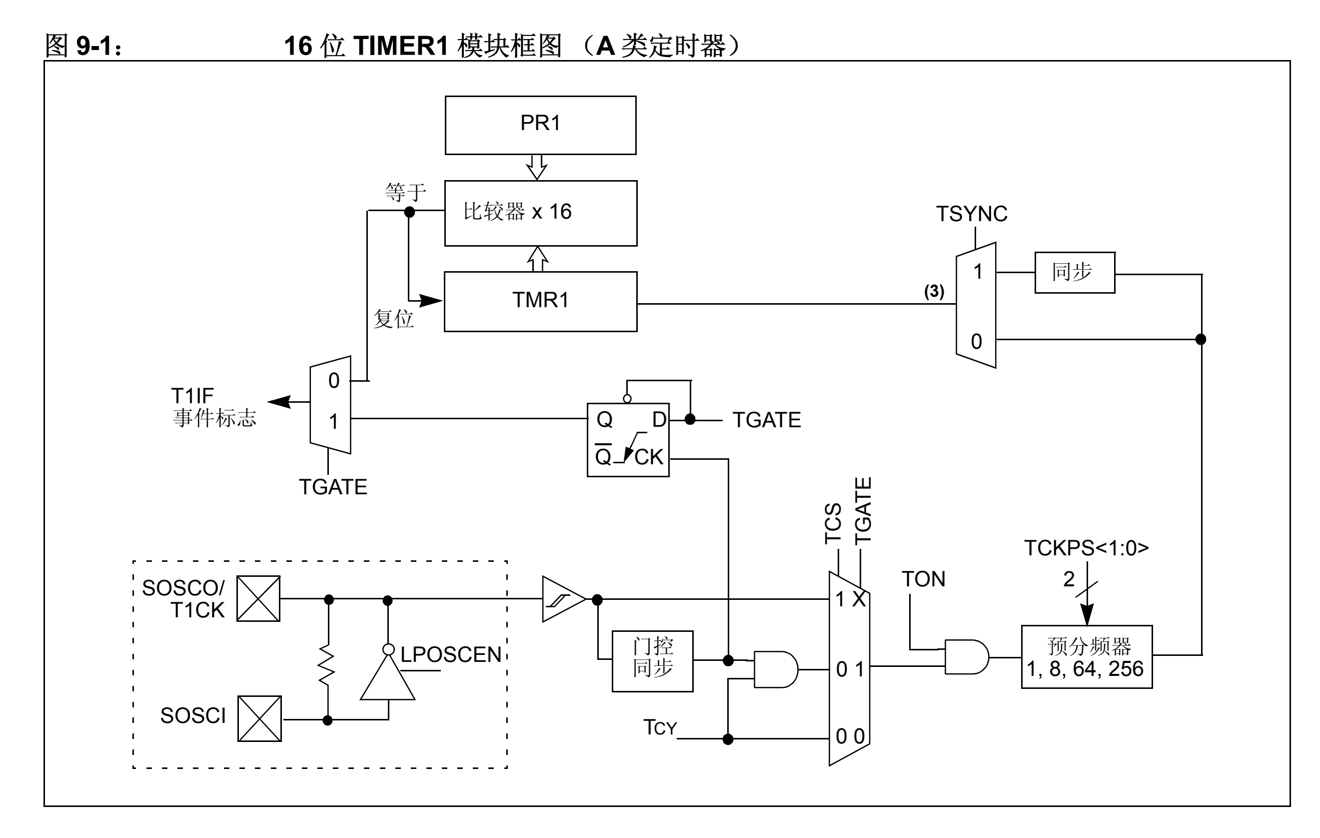
16 位同步计数器模式：在 16 位同步计数器模式下，定时器将在外部时钟信号的上升沿递增，外部时钟与内部时钟同步。定时器计数，直到等于 PR1 中预先装入的值，然后复位至 0，继续计数。

当 CPU 进入空闲模式时，定时器将停止递增，除非对应的 TSIDL 位 = 0。如果 TSIDL = 1，则定时器模块逻辑将在 CPU 空闲模式结束后继续递增。

16 位异步计数器模式：在 16 位异步计数器模式下，定时器将在外部时钟信号的上升沿递增。定时器计数，直到等于 PR1 中预先装入的值，然后复位至 0，继续计数。

当定时器配置为异步工作模式时，且 CPU 进入空闲模式的话，如果 TSIDL = 1，则定时器将停止递增。

图 9-1： 16 位 TIMER1 模块框图 （A 类定时器）



9.1 定时器门控操作

16 位定时器可以置为门控时间累加模式 （Gated Time Accumulation mode） 。该模式使得当门控输入信号 （T1CK 引脚）为高电平时，内部 TCY 能够递增相应的定时器。要使能该模式，必须把控制位 TGATE 置 1 （T1CON<6>）。必须使能定时器（TON = 1），且定时器时钟源设置为外部时钟源 （TCS = 0）。

当 CPU 进入空闲模式时，定时器将停止递增，除非 TSIDL = 0。如果 TSIDL = 1，则定时器将在 CPU 空闲模式终止后继续递增。

9.2 定时器预分频器

16 位定时器的输入时钟 （Fosc/4 或外部时钟）有如下预分频选择：1:1、 1:8、 1:64 或 1:256 ；通过控制位 TCKPS<1:0> （T1CON<5:4>）来选择预分频比。当出现下列事件之一时，预分频器的计数器将清零：

- 写 TMR1 寄存器
- 清零 TON 位 （T1CON<15>）
- 器件复位，如 POR 和 BOR

但是，如果定时器被禁止 （TON = 0），那么定时器的预分频器将不会复位，因为预分频器的时钟停止了。

当写 T1CON 时，TMR1 不会被清零。当写 TMR1 寄存器时， TMR1 将被清零。

9.3 休眠模式下的定时器操作

在 CPU 休眠模式期间，定时器将继续工作，如果：

- 定时器模块被使能 （TON = 1），且
- 定时器时钟源选用外部时钟 （TCS = 1），且
- TSYNC 位 （T1CON<2>）为逻辑 0，这将外部时钟源定义为异步的。

如果所有这三个条件都得到满足，定时器将继续计数，直到等于周期寄存器中的值，然后复位至 0x0000。

当定时器与周期寄存器之间匹配时，如果相应的定时器中断允许位置 1 的话，将产生中断。

9.4 定时器中断

16 位定时器具有在周期匹配时产生中断的能力。当定时器计数与周期寄存器相匹配时， T1IF 位将置 1，如果允许中断的话将产生中断。T1IF 位必须在软件中清除。定时器中断标志 T1IF 位于中断控制器的 IFS0 控制寄存器中。

当使能了门控时间累加模式时，还将在门控信号的下降沿（累加周期的末尾）产生中断。

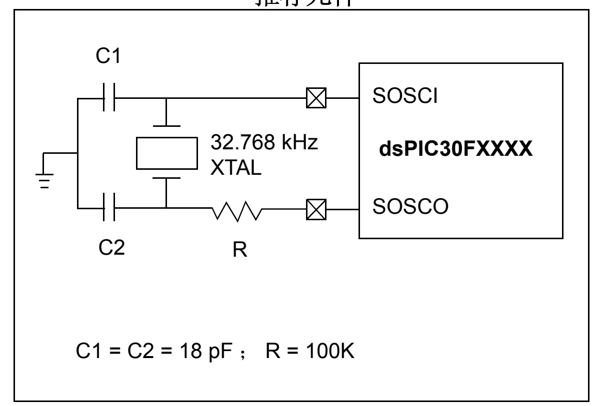
通过相应的定时器中断允许位T1IE来允许定时器中断。定时器中断允许位位于中断控制器的 IEC0 控制寄存器中。

9.5 实时时钟

当 Timer1 工作在实时时钟（RTC）模式下时，提供了当天时间和事件时间标记的能力。关键的 RTC 工作特性如下：

- 使用 32 kHz LP 振荡器工作
- 8 位预分频器
- 低功耗
- 实时时钟中断
- 工作模式，由 T1CON 控制寄存器中的相应位决定。

图 9-2： TIMER1 LP 振荡器 RTC 的推荐元件



9.5.1 RTC 振荡器操作

当 TON = 1、 TCS = 1 且 TGATE = 0 时，定时器在 32 kHz LP 振荡器输出信号的上升沿递增，一直递增到周期寄存器中指定的值，然后复位至 0。

为确保正确工作， TSYNC 位必须置为逻辑 0（异步模式）。

使能 LPOSCEN（OSCCON<1>）将禁止通常的定时器和计数器模式而使能定时器溢出唤醒事件。

当 CPU 进入休眠模式时，如果 32 kHz 外部晶体振荡器处于工作状态、且控制位没有改变的话， RTC 将继续工作。要使 RTC 在空闲模式下继续工作， TSIDL 位应当置为 0。

9.5.2 RTC 中断

当中断事件发生时，相应的中断标志 T1IF 将置 1，如果允许中断的话就将产生中断。 T1IF 位必须在软件中清零。相应的定时器中断标志T1IF位于中断控制器的 IFS0 状态寄存器中。

通过相应的中断允许位 T1IE 来允许中断。定时器中断允许位位于中断控制器的 IEC0 控制寄存器中。

表 9-1： TIMER1 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
TMR1	0100	Timer 1 寄存器																uuuu uuuu uuuu uuuu
PR1	0102	周期寄存器 1																1111 1111 1111 1111
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—	0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

10.0 TIMER2/3 模块

> **注：**本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

本章介绍了 32 位通用定时器模块 （Timer2/3）以及相关的工作模式。图 10-1 给出了 32 位 Timer2/3 模块的简化框图。图 10-2 和图 10-3 分别为将 Timer2/3 配置成的两个独立 16 位定时器 Timer2 和 Timer3 的框图。

> **注：** Timer2 是 B 类定时器而 Timer3 是 C 类定时器。请参见本文挡**第 22.0 节"电气特性"**中相应类型定时器的规范。

Timer2/3 模块是 32 位定时器（可配置为两个 16 位定时器），具有可选择的工作模式。这些定时器为其他外设模块所使用，例如：

- 输入捕捉
- 输出比较 / 简单 PWM

对于定时器的工作模式，下面的各小节将详细说明其设置和控制寄存器以及相应的框图。

32 位定时器有下列模式：

- 两个独立的 16 位定时器（Timer2 和 Timer3），支持所有 16 位工作模式 （异步计数器模式除外）
- 一个 32 位定时器
- 一个 32 位同步计数器

此外，支持如下工作特性：

- ADC 事件触发信号
- 定时器门控操作
- 可选择的预分频比设定
- 空闲和休眠模式下的定时器操作
- 在 32 位周期寄存器匹配时产生中断

通过设定 16 位 T2CON 和 T3CON SFR 中的相应位来决定这些工作模式。

对于 32 位定时器 / 计数器操作， Timer2 是 32 位定时器的低位字，而 Timer3 是高位字。

> **注：** 对于 32 位定时器操作， T3CON 控制位将被忽略。设置和控制只使用 T2CON 控制位 。对于 32 位定时器模块，使用的是 Timer2 时钟和门控输入，但是，中断产生由 Timer3 中断标志位 （T3IF）反映，且中断通过 Timer3 中断允许位（T3IE）来允许。

16 位定时器模式：在 16 位模式下，Timer2 和 Timer3 可以配置为两个独立的 16 位定时器。每个定时器均可设置为 16 位定时器模式或 16 位同步计数器模式。这两种工作模式的细节，请参见**第 9.0 节 "Timer1 模块"**。

Timer2 和 Timer3 之间唯一的功能差异是，Timer2 提供了时钟预分频器输出同步。这对于高频外部时钟输入很有用。

32 位定时器模式：在 32 位定时器模式下，定时器在每个指令周期递增，一直递增到与预先装入组合 32 位周期寄存器PR3/PR2中的值匹配，然后复位至0，继续计数。

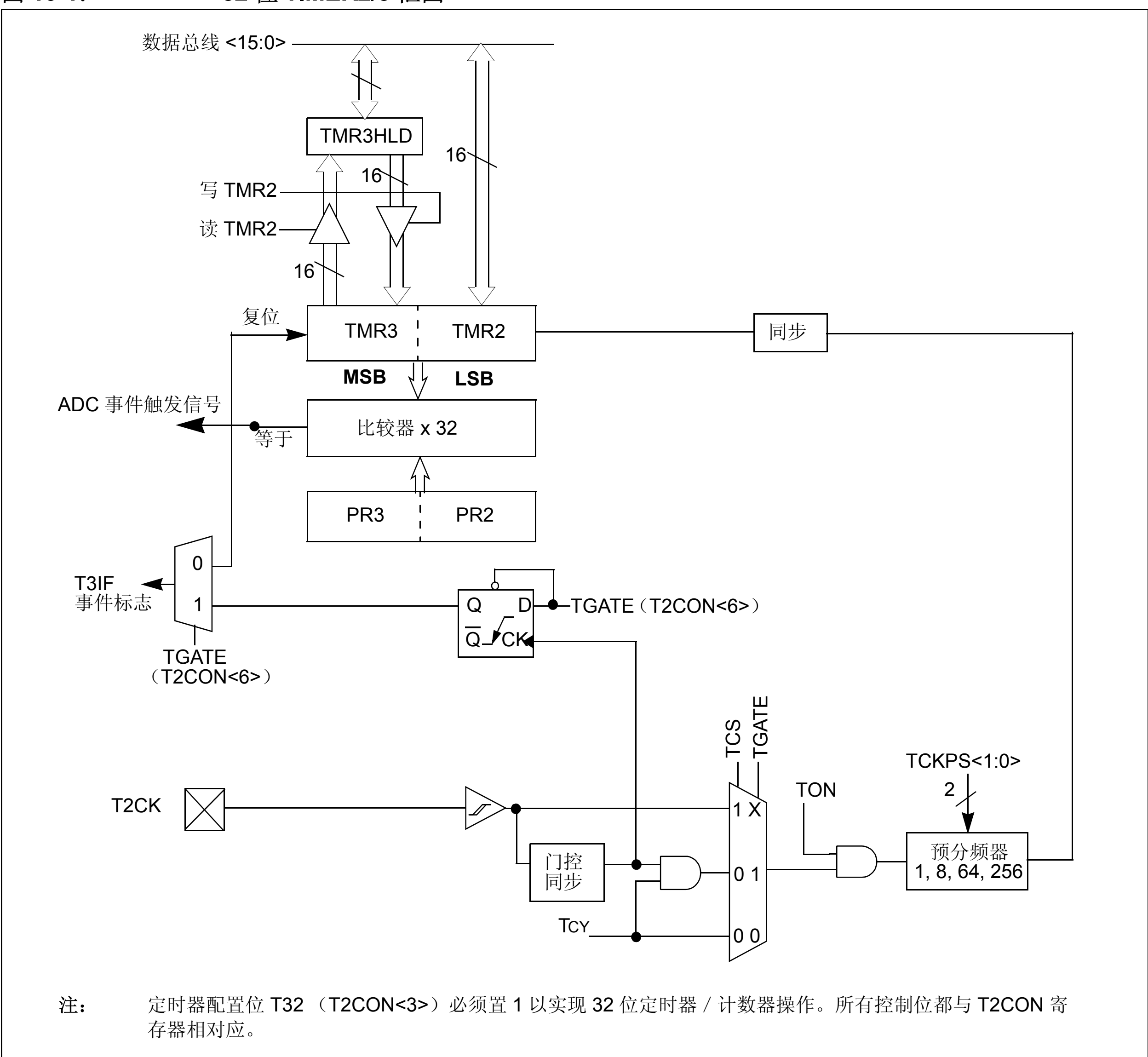
对于 Timer2/Timer3 对的同步 32 位读操作，读低位字（TMR2 寄存器）将导致高位字 （msw）被读取并锁存到一个 16 位保持寄存器 （称作 TMR3HLD）。

对于同步 32 位写操作，必须首先写入保持寄存器（TMR3HLD）。如果后续操作是写 TMR2 寄存器的话，则 TMR3HLD 的内容将传送并锁存到 32 位定时器（TMR3）的 MSB 中。

32 位同步计数器模式：在 32 位同步计数器模式下，定时器将在外部时钟信号的上升沿递增，外部时钟信号与内部时钟同步。定时器计数，一直递增到与预先装入组合 32 位周期寄存器 PR3/PR2 中的值匹配，然后复位至 0，继续计数。

当定时器配置为同步计数器工作模式时，且 CPU 进入空闲模式的话，定时器将停止递增，除非 TSIDL（T2CON<13>）位 = 0。如果 TSIDL = 1，定时器模块逻辑将继续递增，直到 CPU 空闲模式终止。

图 10-1： 32 位 TIMER2/3 框图



注： 定时器配置位 T32 （T2CON<3>）必须置 1 以实现 32 位定时器 / 计数器操作。所有控制位都与 T2CON 寄存器相对应。

图 10-2： 16 位 TIMER2 框图 （ B 类定时器）

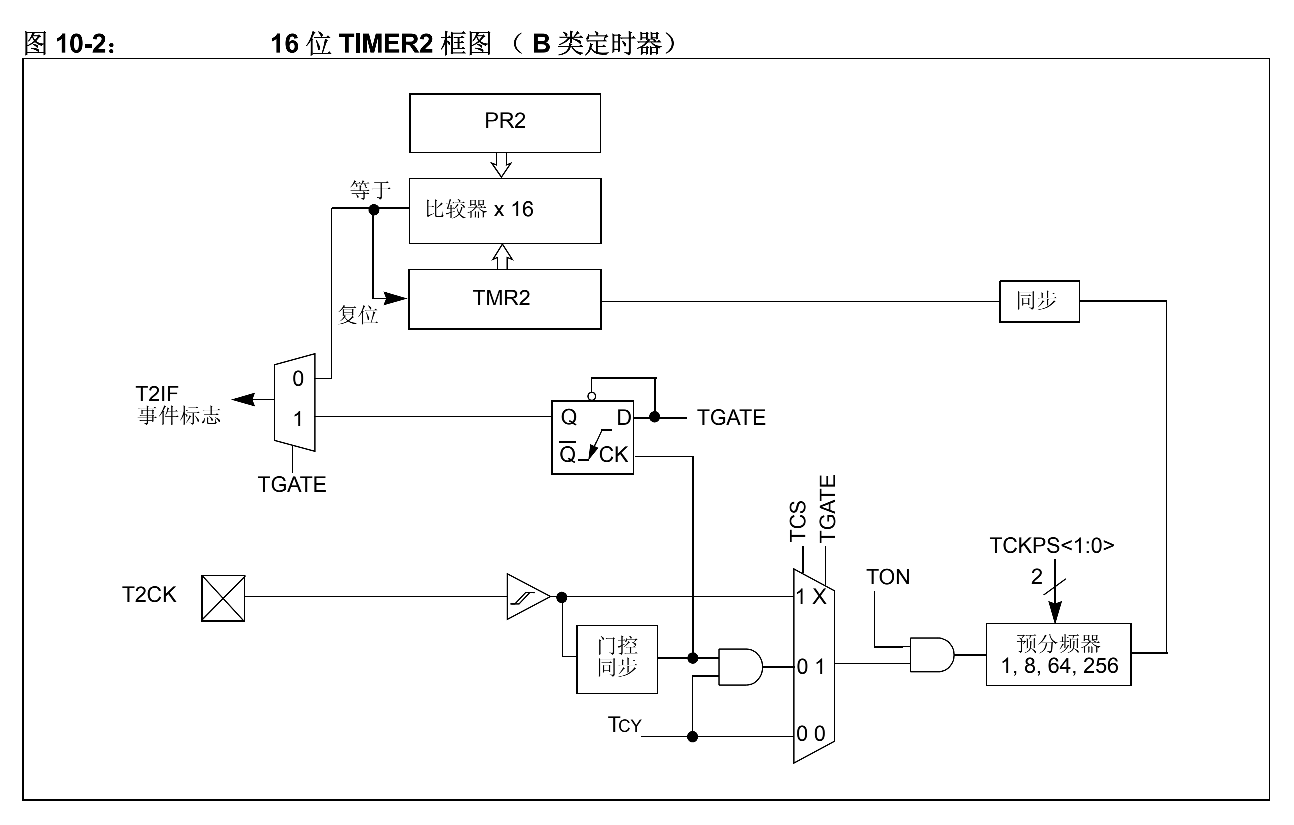
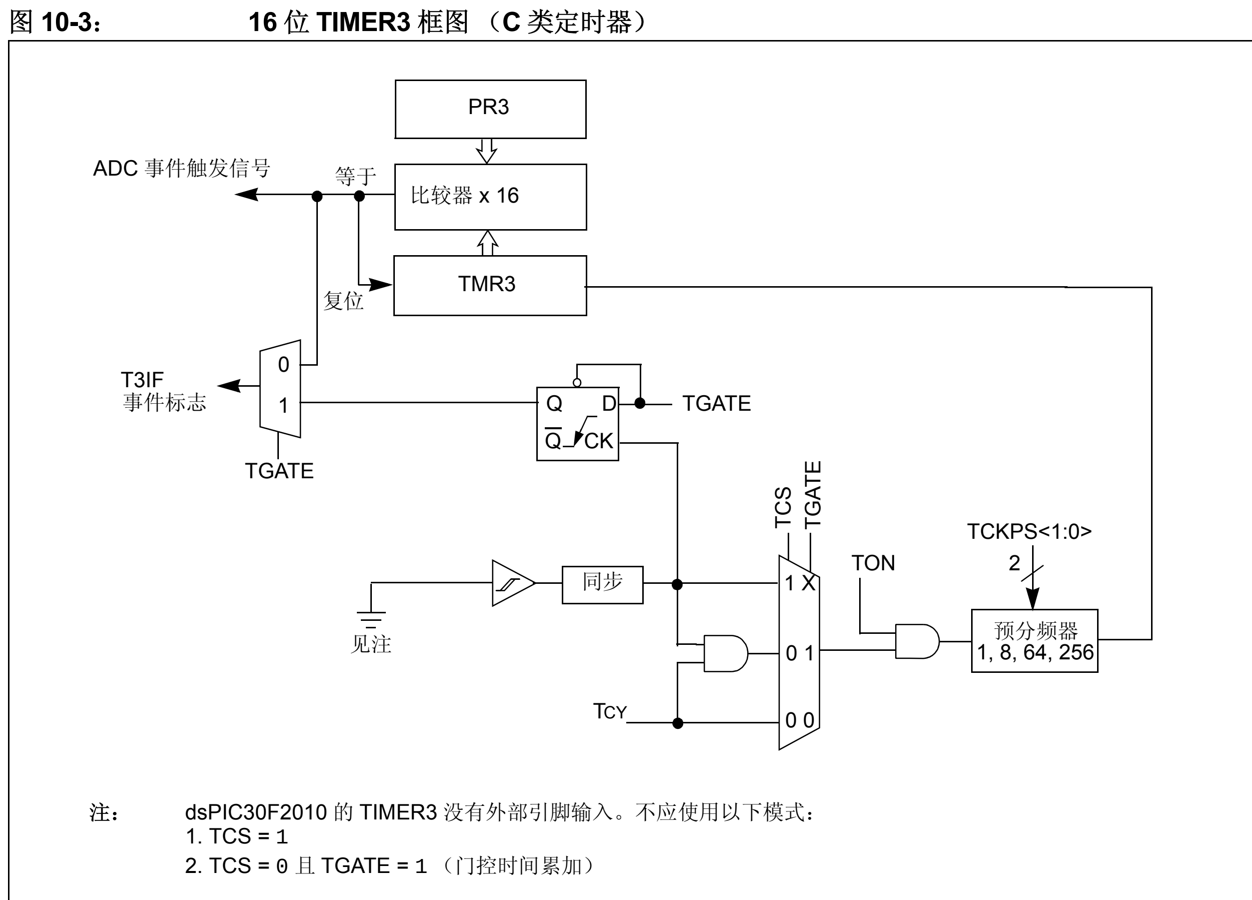


图 10-3： 16 位 TIMER3 框图 （C 类定时器）



注： dsPIC30F2010 的 TIMER3 没有外部引脚输入。不应使用以下模式：
1. TCS = 1
2. TCS = 0 且 TGATE = 1 （门控时间累加）

10.1 定时器门控操作

32 位定时器可以置为门控时间累加模式。该模式使得当门控输入信号（T2CK 引脚）为高电平时，内部 TCY 能够递增相应的定时器。要使能该模式，必须置 1 控制位 TGATE （T2CON<6>）。在此模式下， Timer2 是发起时钟源。对于 Timer3， TGATE 设定被忽略。必须使能定时器 （TON = 1），且定时器时钟源设置为内部时钟源 （TCS = 0）。

外部信号的下降沿终止计数操作，但不会复位定时器。要从零开始计数，用户必须复位定时器。

10.2 ADC 事件触发信号

当 32 位定时器（TMR3/TMR2）与 32 位组合周期寄存器 （PR3/PR2）匹配时， Timer3 将生成 ADC 特殊事件触发信号。

10.3 定时器预分频器

定时器的输入时钟 （FOSC/4 或外部时钟），有 1:1、1:8、 1:64 或 1:256 的预分频选择；通过控制位 TCKPS<1:0>（T2CON<5:4> 和 T3CON<5:4>）来选择预分频比。对于 32 定时器操作，发起时钟源是 Timer2。在此模式下不能进行 Timer3 的预分频器操作。当出现下列事件之一时，预分频器的计数器将清零：

- 写 TMR2/TMR3 寄存器
- 清零任一 TON 位（T2CON<15> 或 T3CON<15>）
- 器件复位，如 POR 和 BOR

但是，如果定时器被禁止（TON = 0），那么 Timer2 预分频器将不会被复位，因为预分频器的时钟停止了。

当写 T2CON/T3CON 时， TMR2/TMR3 不会被清零。

10.4 休眠模式下的定时器操作

在 CPU 休眠模式期间，定时器不工作，因为内部时钟被禁止。

10.5 定时器中断

32 位定时器模块具有在周期匹配时，或者在外部门控信号下降沿产生中断的能力。当 32 位定时器计数与相应的周期寄存器相匹配时，或者检测到外部 “门控”信号的下降沿时， T3IF 位 （IFS0<7>）将置 1，如果允许中断的话则将产生中断。在此模式下，T3IF 中断标志用作中断源。 T3IF 位必须在软件中清零。

通过相应的定时器中断允许位 T3IE （IEC0<7>）来允许中断。

表 10-1： TIMER2/3 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
TMR2	0106	Timer2 寄存器																uuuu uuuu uuuu uuuu
TMR3HLD	0108	Timer3 保持寄存器 （仅适用于 32 位定时器操作）																uuuu uuuu uuuu uuuu
TMR3	010A	Timer3 寄存器																uuuu uuuu uuuu uuuu
PR2	010C	周期寄存器 2																1111 1111 1111 1111
PR3	010E	周期寄存器 3																1111 1111 1111 1111
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	T32	—	TCS	—	0000 0000 0000 0000
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

11.0 输入捕捉模块

> **注：**本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

本章对输入捕捉模块和相关的工作模式进行了介绍。此模块提供的工作特性可用于需要进行频率（周期）和脉冲测量的应用场合。图 11-1 给出了输入捕捉模块的框图。输入捕捉功能可用于以下模式：

- 频率 / 周期 / 脉冲测量
- 其他外部中断源

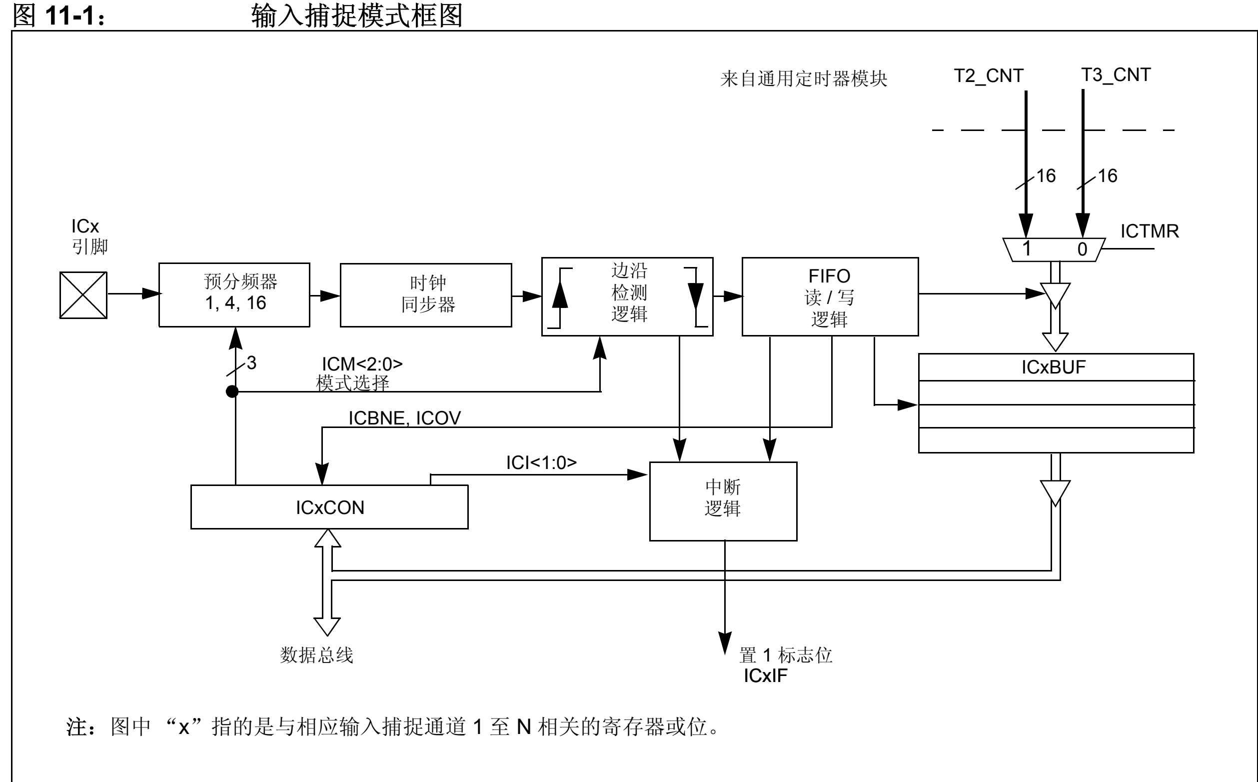
输入捕捉模块的关键工作特性是：

- 简单捕捉事件模式
- Timer2 和 Timer3 模式选择
- 输入捕捉事件发生时产生中断

通过在 ICxCON 寄存器 （其中， x = 1,2,...,N）中设置相应的位，选择上述工作模式。 dsPIC DSC 器件具有多达 8 个捕捉通道 （即， N 的最大值为 8）。

> **注：** dsPIC30F2010 器件具有 4 个捕捉输入——IC1、IC2、IC7 和 IC8。这 4 个捕捉通道的命名意在保持与其他 dsPIC DSC 器件的软件兼容性。

图 11-1： 输入捕捉模式框图



注：图中 “x” 指的是与相应输入捕捉通道 1 至 N 相关的寄存器或位。

11.1 简单捕捉事件模式

dsPIC30F 产品系列中的简单捕捉事件是：

- 在每个下降沿进行捕捉
- 在每个上升沿进行捕捉
- 在每第 4 个上升沿进行捕捉
- 在每第 16 个上升沿进行捕捉
- 在每个上升沿和下降沿进行捕捉

通过设置适当的位 ICM<2:0> （ICxCON<2:0>），选择上述简单输入捕捉模式。

11.1.1 捕捉预分频器

有四种输入捕捉预分频器设置，它们由位 ICM<2:0> （ICxCON<2:0>）指定。每当捕捉通道关闭时，预分频器计数器将清零。另外，任何复位都将清零预分频器计数器。

11.1.2 捕捉缓冲器操作

每个捕捉通道都有一个关联的 FIFO 缓冲器，缓冲器宽16位，深度为四级。有两个状态标志，提供FIFO缓冲器的状态：

- ICBNE——输入捕捉缓冲器非空
- ICOV—— 输入捕捉溢出

ICBFNE将在第一个输入捕捉事件发生时被置1，并且一直保持置 1 状态，直到所有的捕捉事件都已从 FIFO 中读出。每次从 FIFO 中读出一个字，余下的字将在缓冲器中前进一个位置。

当 FIFO 已满 （FIFO 中有四个捕捉事件）时，如果在读 FIFO 之前发生第五个捕捉事件，则将产生溢出，且 ICOV 位将被设置成逻辑 1。第五个捕捉事件将丢失，不会存入 FIFO。而且在从缓冲器读出全部四个事件之前不会再捕捉任何事件。

如果 FIFO 已空、且还没有收到新的捕捉事件，这时读 FIFO 将产生不确定的结果。

11.1.3 TIMER2 和 TIMER3 选择模式

输入捕捉模块有多达 8 个输入捕捉通道。每个通道均可选择两个定时器 （Timer2 和 Timer3）之一作为时基。

通过特殊功能寄存器位 ICTMR （ICxCON<7>），来进行定时器资源的选择。对于输入捕捉模块， Timer3 是缺省的可用定时器资源。

11.1.4 霍尔传感器模式

当输入捕捉模块设置为在每个边沿 （上升沿和下降沿，ICM<2:0> = 001）进行捕捉时，输入捕捉逻辑将执行下列操作：

- 输入捕捉中断标志在每个边沿 （上升沿和下降沿）置 1。
- 捕捉时产生中断模式设置位 ICI<1:0> 被忽略，因为每次捕捉都产生中断。
- 在这个模式下，不产生捕捉溢出。

11.2 休眠和空闲模式期间的输入捕捉工作

当器件在 CPU 空闲或休眠模式下时，输入捕捉事件将使器件唤醒。如果允许中断，还将产生中断。

如果 ICM<2:0> = 111 且中断允许位置 1，那么当捕捉事件发生时，输入捕捉模块将使 CPU 从休眠或空闲模式中唤醒，这与使能的时钟无关。如果处理中断的条件已经得到满足的话，前述的唤醒还将产生中断。唤醒特性可以用来增加额外的外部引脚中断。

11.2.1 CPU 休眠模式下的输入捕捉

CPU 休眠模式允许输入捕捉模块以功能受到限制的方式工作。在 CPU 休眠模式下， ICI<1:0> 位不适用，输入捕捉模块只能用作外部中断源。

当器件处于休眠模式下时，为了使用输入捕捉模块，必须把捕捉模块设置成仅在上升沿产生中断 （ICM<2:0> = 111）。在此模式下，4:1 或 16:1 的预分频比设定不适用。

11.2.2 CPU 空闲模式下的输入捕捉

CPU 空闲模式允许输入捕捉模块以功能受到限制的方式工作。在 CPU 空闲模式下，通过 ICI<1:0> 位选择的中断模式是适用的，由控制位 ICM<2:0> 定义的 4:1 和 16:1 捕捉预分频比设定同样也适用。这个模式要求使能所选择的定时器。此外， ICSIDL 位必须置为逻辑 0。

在 CPU 空闲模式下，如果输入捕捉模块定义为 ICM<2:0> = 111，则输入捕捉引脚仅用作外部中断引脚。

11.3 输入捕捉中断

输入捕捉通道具有根据选定的捕捉事件数来产生中断的能力。通过控制位 ICI<1:0> （ICxCON<6:5>）来设定选定的捕捉事件数。

每个通道都有一个中断标志位（ICxIF）。各捕捉通道中断变址位于相应的 IFSx 状态寄存器中。

通过相应捕捉通道中断允许位 （ICxIE）来允许中断；捕捉通道中断允许位位于相应的 IEC 控制寄存器中。

表 **11-1**： 输入捕捉寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
IC1BUF	0140	输入捕捉 1 寄存器																uuuu uuuu uuuu uuuu
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC2BUF	0144	输入捕捉 2 寄存器																uuuu uuuu uuuu uuuu
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC3BUF	0148	输入捕捉 3 寄存器																uuuu uuuu uuuu uuuu
IC3CON	014A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC4BUF	014C	输入捕捉 4 寄存器																uuuu uuuu uuuu uuuu
IC4CON	014E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC5BUF	0150	输入捕捉 5 寄存器																uuuu uuuu uuuu uuuu
IC5CON	0152	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC6BUF	0154	输入捕捉 6 寄存器																uuuu uuuu uuuu uuuu
IC6CON	0156	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC7BUF	0158	输入捕捉 7 寄存器																uuuu uuuu uuuu uuuu
IC7CON	015A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000
IC8BUF	015C	输入捕捉 8 寄存器																uuuu uuuu uuuu uuuu
IC8CON	015E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>		ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

12.0 输出比较模块

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

本章对输出比较模块和相关工作模式进行了介绍。此模块提供的特性适用于需要以下工作模式的应用：

- 产生可变宽度输出脉冲
- 功率因数校正

图 12-1 给出了输出比较模块的框图。

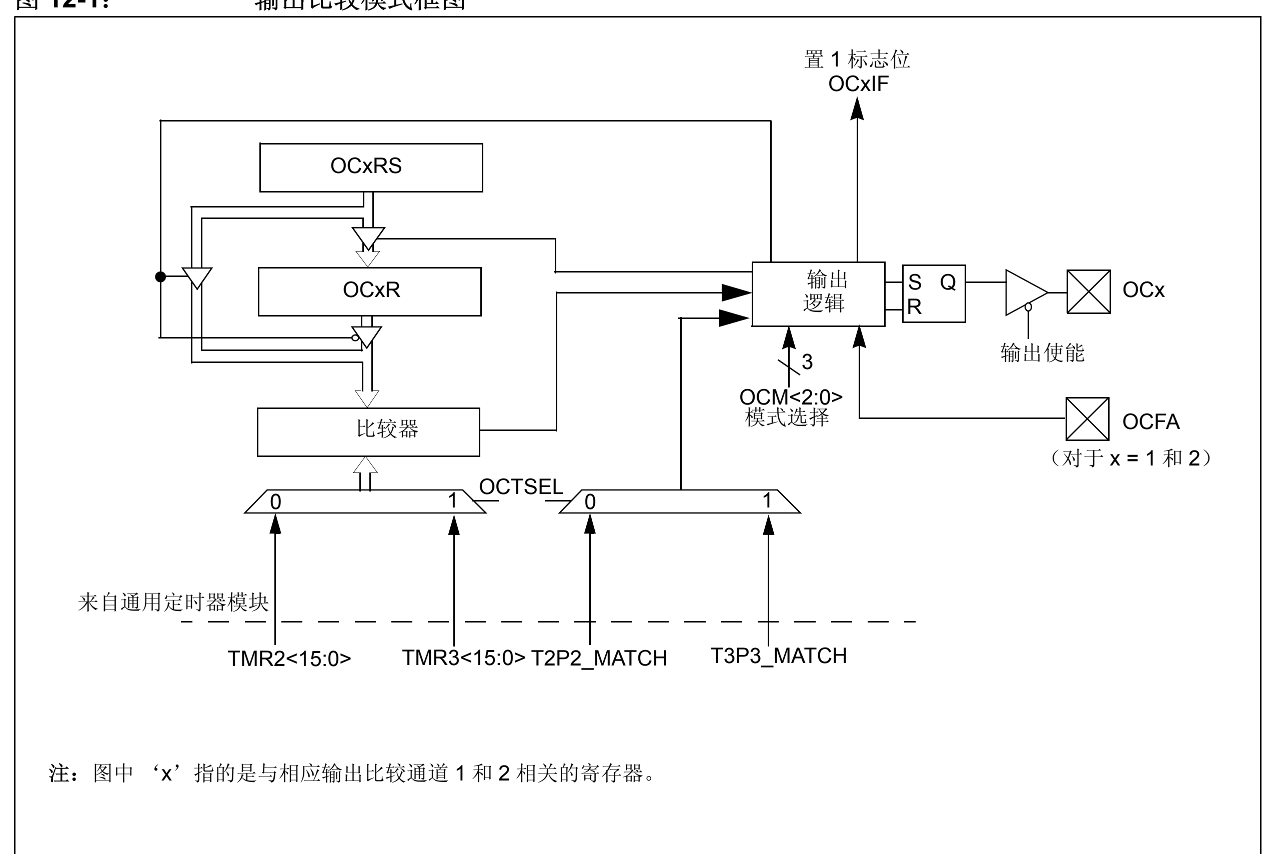
输出比较模块的关键工作特性包括：

- Timer2 和 Timer3 模式选择
- 简单输出比较匹配模式
- 双输出比较匹配模式
- 简单 PWM 模式
- 休眠和空闲模式下的输出比较
- 发生输出比较 /PWM 事件时将产生中断

通过设定 16 位 OCxCON SFR（其中 x = 1 或 2）中相应的位可确定这些操作模式。

图中的 OCxRS 和 OCxR 表示双比较寄存器。在双比较模式下，OCxR 寄存器用于第一次比较而 OCxRS 用于第二次比较。

图 12-1： 输出比较模式框图



注：图中 ‘x’ 指的是与相应输出比较通道 1 和 2 相关的寄存器。

12.1 Timer2 和 Timer3 模式选择

每个输出比较通道均可选择两个 16 位定时器 Timer2 和 Timer3 中的一个。

通过 OCTSEL 位 （OCxCON<3>）来控制定时器的选择。 Timer2 是输出比较模块缺省的定时器资源。

12.2 简单输出比较匹配模式

当控制位 OCM<2:0> （OCxCON<2:0>） = 001、 010 或 011 时，所选输出比较通道被配置为下列三种简单输出比较匹配模式之一：

- 比较强制 I/O 引脚为低电平
- 比较强制 I/O 引脚为高电平
- 比较使 I/O 引脚状态翻转

上述模式中使用了 OCxR 寄存器。OCxR 寄存器将装入一个值，然后与所选的递增定时器的计数值作比较。进行比较时，将出现上述比较匹配模式之一。如果在达到 OCxR 中的值之前，计数器复位为零，则 OCx 引脚的状态保持不变。

12.3 双输出比较匹配模式

当控制位 OCM<2:0>（OCxCON<2:0>）= 100 或 101 时，所选输出比较通道被配置为下列两种双输出比较匹配模式之一：

- 单输出脉冲模式
- 连续输出脉冲模式

12.3.1 单脉冲模式

用户将模块配置为生成单输出脉冲，需要执行下列步骤（假设定时器已关闭）：

- 确定指令周期时间 T_{CY}。
- 根据 T_{CY} 计算所需的脉冲宽度值。
- 计算从定时器起始值 0x0000 到脉冲开始的时间。
- 把脉冲宽度开始和停止时间分别写入 OCxR 和 OCxRS 比较寄存器 （x 表示通道 1 或 2）。
- 设置定时器周期寄存器的值，使之等于或大于 OCxRS 比较寄存器中的值。
- 设置 OCM<2:0> = 100。
- 使能定时器， TON （TxCON<15>） = 1。

要启动另一个单脉冲，进行另一次写操作设置 OCM<2:0> = 100 即可。

12.3.2 连续脉冲模式

将模块配置为生成连续输出脉冲，则需要执行下列步骤：

- 确定指令周期时间 T_{CY}。
- 根据 T_{CY} 计算所需的脉冲宽度值。
- 计算从定时器起始值 0x0000 到脉冲开始的时间。
- 把脉冲宽度开始和停止时间分别写入 OCxR 和 OCxRS 比较寄存器 （x 表示通道 1 或 2）。
- 设置定时器周期寄存器的值，使之等于或大于 OCxRS 比较寄存器中的值。
- 设置 OCM<2:0> = 101。
- 使能定时器， TON （TxCON<15>） = 1。

12.4 简单 PWM 模式

当控制位 OCM<2:0>（OCxCON<2:0>）= 110 或 111 时，所选输出比较通道配置为 PWM 工作模式。当配置为 PWM 工作模式时， OCxR 是主锁存器 （只读）， OCxRS是辅助锁存器。这样能使PWM 信号跳变时不会产生毛刺。

为了把输出比较模块配置为 PWM 工作模式，用户必须执行下列步骤：

1. 写适当的周期寄存器，设置 PWM 周期。
2. 写 OCxRS 寄存器，设置 PWM 占空比。
3. 把输出比较模块配置为 PWM 工作模式。
4. 设置 TMRx 预分频比，使能定时器， TON （TxCON<15>） = 1。

12.4.1 PWM 的输入引脚故障保护

当控制位 OCM<2:0>（OCxCON<2:0>）= 111 时，所选输出比较通道还是配置为 PWM 工作模式，但具有输入故障保护的附加功能。在此模式下，如果在 OCFA/B 引脚上检测到逻辑 0，则对应的 PWM 输出引脚将被置于高阻输入状态。 OCFLT 位 （OCxCON<4>）表明是否产生故障条件。此状态将保持，直到下列事件都已发生：

t 外部故障条件已经消除。
- 通过写适当的控制位，重新使能了 PWM 模式。

12.4.2 PWM 周期

通过写入 PRx 寄存器可以指定 PWM 周期。可使用公式 12-1 计算 PWM 周期。

公式 12-1： PWM 周期

PWM 周期 = [(PRx) + 1] • 4 • TOSC •（TMRx 预分频比）

PWM 频率定义为 1/[PWM 周期]。

当所选 TMRx 等于其对应的周期寄存器 PRx 时，在下一个递增周期将发生如下四个事件：

- TMRx 清零。
- OCx 引脚置 1。
  - 例外 1：如果 PWM 占空比为 0x0000，则 OCx 引脚将保持为低电平。
  - 例外 2：如果占空比大于 PRx，则引脚将保持为高电平。
- 把 PWM 占空比从 OCxRS 锁存到 OCxR 中。
- 相应的定时器中断标志置 1。
- 参见图 12-1 的关键 PWM 周期比较。为了清晰起见，图中引用的是 Timer3 。

12.5 CPU 休眠模式期间的输出比较工作

当CPU进入休眠模式时，所有的内部时钟都将停止。从而，输出比较通道将把引脚驱动为 CPU 进入休眠模式之前的有效状态。

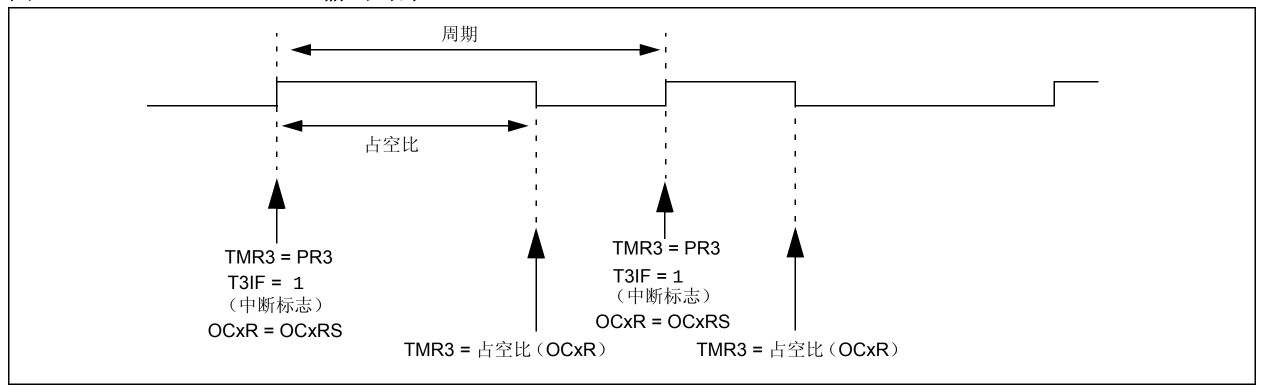
例如，当 CPU 进入休眠状态时，如果引脚为高电平，则引脚将继续保持为高电平。类似地，当 CPU 进入休眠状态时，如果引脚为低电平，它将继续保持为低电平。在这两种情况下，当器件唤醒时，输出比较模块都将恢复工作。

12.6 CPU 空闲模式期间的输出比较工作

当 CPU 进入空闲模式时，输出比较模块仍然能够全功能工作。

如果 OCSIDL 位（OCxCON<13>）为逻辑 0，所选时基（Timer2 或 Timer3）使能且所选定时器的 TSIDL 位为逻辑 0，输出比较通道将在 CPU 空闲模式期间工作。

图 12-2： PWM 输出时序



12.7 输出比较中断

输出比较通道具有在比较匹配时产生中断的能力，而与选择的匹配模式无关。

对于除 PWM 之外的所有模式，当比较事件发生时，相应的中断标志（OCxIF）置 1，如果允许中断的话还将产生中断。OCxIF 位位于相应的 IFS 状态寄存器中，它必须用软件清零。通过相应的比较中断允许位（OCxIE）来允许中断； OCxIE 位位于相应的 IEC 控制寄存器中。

对于 PWM 模式，当事件发生时，对应的定时器中断标志位（T2IF 或 T3IF）置 1，如果允许中断的话还将产生中断。TxIF 位在 IFS0 状态寄存器中，它的清零必须在软件中进行。通过位于 IEC0 控制寄存器中的定时器中断允许位（T2IE 或 T3IE），来允许相应的中断。在 PWM 工作模式下，输出比较中断标志始终不会置 1。

表 **12-1**：　输出比较寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
OC1RS	0180	输出比较 1 主寄存器																0000 0000 0000 0000
OC1R	0182	输出比较 1 从寄存器																0000 0000 0000 0000
OC1CON	0184	—	OCFRZ	OCSIDL	—	—	—	—	—	—	—	—	OCFLT1	OCTSEL1	OCM<2:0>			0000 0000 0000 0000
OC2RS	0186	输出比较 2 主寄存器																0000 0000 0000 0000
OC2R	0188	输出比较 2 从寄存器																0000 0000 0000 0000
OC2CON	018A	—	OCFRZ	OCSIDL	—	—	—	—	—	—	—	—	OCFLT2	OCTSEL2	OCM<2:0>			0000 0000 0000 0000

图注：　u = 未初始化位

注：　有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

13.0 正交编码器接口（QEI）模块

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

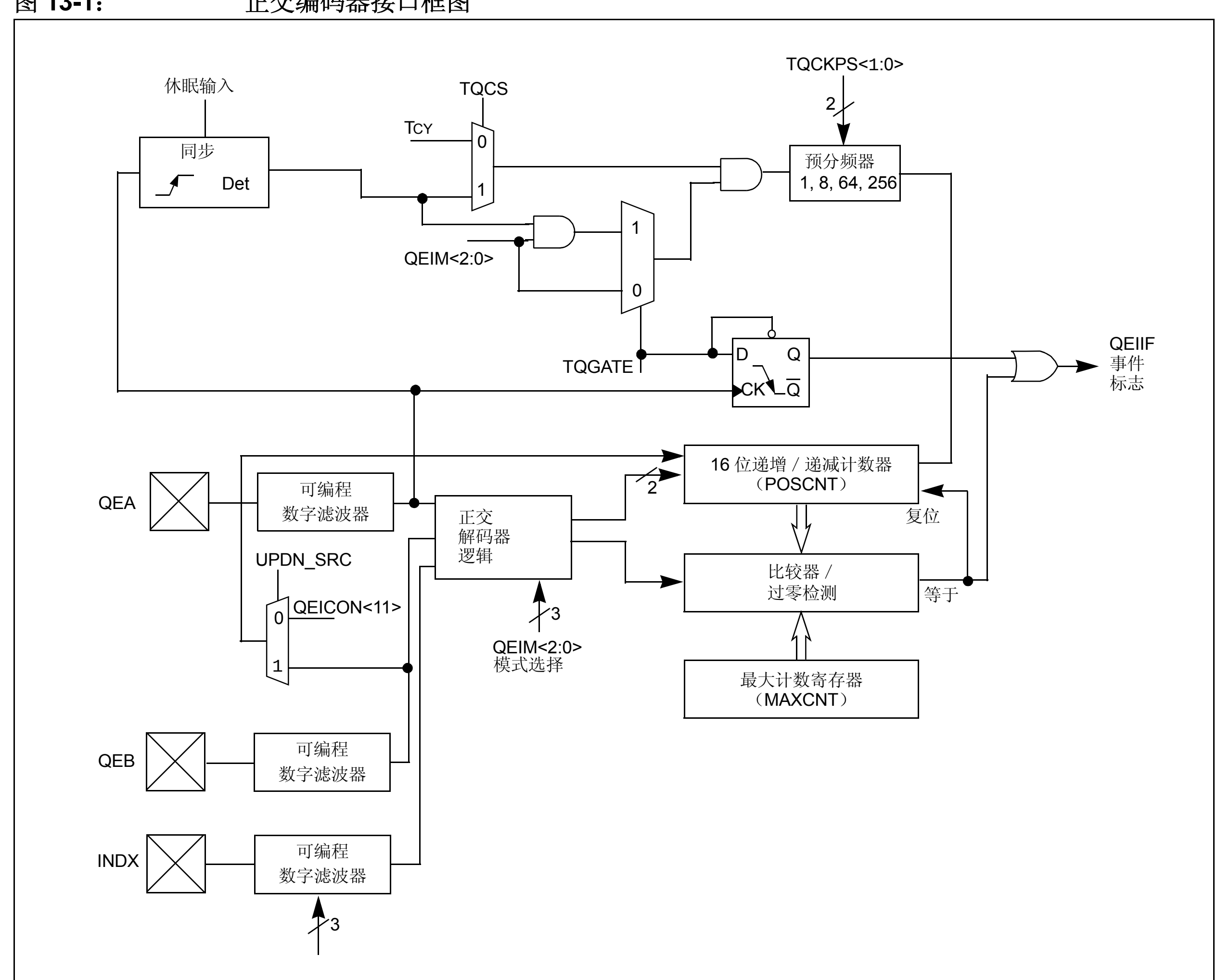
本章介绍了正交编码器接口（Quadrature Encoder Interface，QEI）模块和相关的工作模式。QEI 模块提供了与增量式编码器的接口以获得电机位置数据。增量式编码器在电机控制应用中是非常有用的。

正交编码器接口（QEI）是多种电机控制应用中不可或缺的关键部分，例如开关磁阻（Switched Reluctance，SR）电机和交流感应电机（AC Induction Motor，ACIM）。QEI 的工作特性包括：

- 三个输入通道，分别为两相信号和索引脉冲输入
- 16 位递增 / 递减位置计数器
- 计数方向状态
- 位置测量（x2 和 x4）模式
- 输入端上的可编程数字噪声滤波器
- 备用的 16 位定时器 / 计数器模式
- 正交编码器接口中断

通过设定相应位 QEIM<2:0>（QEICON<10:8>）可确定这些工作模式。图 13-1 给出了正交编码器接口的框图。

图 13-1： 正交编码器接口框图



13.1 正交编码器接口逻辑

典型的增量式编码器（也称为光电式编码器）具有三个输出：A 相、 B 相和索引脉冲。在 ACIM 和 SR 电机的位置和速度控制中，这三个信号非常有用，且通常是必须的。

A 相（QEA）和 B 相（QEB）通道输出信号具有特定的相位关系。如果 A 相超前 B 相，那么认为电机正向旋转。如果 A 相滞后于 B 相，那么认为电机反向旋转。

第三个通道称为索引脉冲，每转一圈产生一个脉冲，作为基准用来确定绝对位置。索引脉冲与 A 相和 B 相一致，皆为低电平。

13.2 16 位递增 / 递减位置计数器模式

16 位递增 / 递减计数器在每一个计数脉冲递增或递减计数，该脉冲信号由 A 相和 B 相输入信号的关系确定。计数器作为一个积分器，其计数值与位置成正比。计数方向由 UPDN 信号确定，该信号由正交编码器接口逻辑产生。

13.2.1 位置计数器错误检查

CNTERR 位（QEICON<15>）提供了 QEI 中的位置计数错误检查功能。只有当位置计数器配置为通过索引脉冲复位模式 （QEIM<2:0> = 110 或 100）时，才能使用错误检查功能。在这些模式下， POSCNT 寄存器中的内容将与值 （0xFFFF 或 MAXCNT + 1，取决于转向）进行比较。如果检测到这些值，将通过置 1 CNTERR 位产生一个错误条件并产生 QEI 计数错误中断。通过置 1 CEID 位（DFLTCON<8>）可禁止 QEI 计数错误中断。在检测到错误之后，位置计数器仍将继续对编码器脉冲边沿进行计数。 POSCNT 寄存器继续递增 / 递减计数直至出现自然计满溢出 / 下溢。发生自然计满溢出 / 下溢事件时不产生中断。CNTERR 位是一个可读 / 写位，由用户采用软件方式对其进行复位。

13.2.2 位置计数器复位

位置计数器复位使能位 POSRES （QEI<2>）用于确定当检测到索引脉冲时是否对位置计数器进行复位。只有当 QEIM<2:0> = ‘100’或‘110’时，该位才适用。

如果 POSRES 位设置为‘1’，则位置计数器在检测到索引脉冲时将被复位。如果 POSRES 位设定为 ‘0’，则当检测到索引脉冲时位置计数器将不会被复位。位置计数器将继续进行递增或递减计数，且在发生计满溢出 / 下溢时进行复位。

当选择使用 INDX 信号对位置计数器（POSCNT）进行复位时，用户必须指定 QEA 和 QEB 输入引脚的状态。只有这些状态匹配时，复位才会发生。通过 DFLTCON<10:9> 寄存器中的 IMV<1:0> 位可选择这些状态。

当 POSCNT 寄存器将被复位时，IMV<1:0>（索引匹配值）位允许用户在索引脉冲期间指定 QEA 和 QEB 输入引脚的状态。

在 4X 正交计数模式下：
IMV1 = 索引脉冲匹配所要求的 B 相输入信号的状态
IMV0 = 索引脉冲匹配所要求的 A 相输入信号的状态

在 2X 正交计数模式下：
IMV1= 为索引状态匹配选择的相输入信号（0 = A 相，1 =B 相）
IMV0= 索引脉冲匹配要求的所选相输入信号的状态

当检测到索引脉冲但位置计数器未发生计满溢出 / 下溢事件时，仍将产生中断。

13.2.3 计数方向状态

如前所述， QEI 逻辑将根据 A 相和 B 相之间的关系产生 UPDN 信号。除输出引脚外，还将此内部 UPDN 信号的状态传送至 SFR 的 UPDN 位 （QEICON<11>），该位是只读位。

> 注：QEI 引脚复用为模拟输入引脚。用户必须确保在 ADPCFG 寄存器中所有 QEI 相关引脚被设定为数字输入。

13.3 位置测量模式

QEI 支持两种测量模式，分别称为 x2 和 x4 模式。通过位于 SFR QEICON<10:8> 中的模式选择位 QEIM<2:0> 可选择模式。

当控制位 QEIM<2:0> = 100 或 101 时，将选择 x2 测量模式，此时 QEI 逻辑将只通过 A 相输入信号来确定位置计数器递增速率。A 相信号的每个上升沿和下降沿都将导致位置计数器递增或递减。和在 x4 模式中一样， B 相信号仍然用于计数器方向的确定。

在 x2 测量模式中，存在导致位置计数器复位的两种不同原因：

1. 检测到索引脉冲，QEIM<2:0> = 100，导致位置计数器复位。
2. 计数值与 MAXCNT， QEIM<2:0> = 101 匹配时导致位置计数器复位。

当控制位 QEIM<2:0> = 110 或 111 时，将选择 x4 测量模式，此时 QEI 逻辑将通过 A 相和 B 相输入信号的每个边沿来确定位置计数器递增速率。A 相和 B 相输入信号的每个边沿都将导致位置计数器递增或递减。

在 x4 测量模式中，存在导致位置计数器复位的两种不同原因：

1. QEIM<2:0> = 110 时，检测到索引脉冲导致位置计数器复位。
2. QEIM<2:0> = 111时，计数值与MAXCNT匹配导致位置计数器复位。

x4 测量模式为确定电机位置提供了更高精度的测量数据 （更多的位置计数）。

13.4 可编程数字噪声滤波器

数字噪声滤波器部分负责抑制输入捕捉或正交信号中的噪声。施密特触发器输入和三时钟周期延迟滤波器共同用来抑制低电平噪声和短时的大噪声尖峰。这些噪声干扰经常出现在易受噪声干扰的应用中，例如电机系统。

该滤波器可以确保在三个连续的滤波器周期内都获得同一个稳定值之后，才允许经过滤波的输出信号发生变化。

可通过位 QECK<2:0> （DFLTCON<6:4>）对 QEA、QEB 和 INDX 引脚的数字滤波器时钟分频频率进行编程设定，该频率来源自基本指令周期 TCY。

QEOUT 位必须为 ‘1’ 以使能 QEA、QEB 和 INDX 通道的滤波器输出功能。在 POR 和 BOR 时，所有通道的滤波器网络将被禁止。

13.5 备用 16 位定时器 / 计数器

当 QEI 模块未被配置为 QEI 模式 QEIM<2:0> = 001 时，此模块可被配置为简单的 16 位定时器 / 计数器。通过 QEICON SFR 寄存器可实现辅助定时器的设定和控制。此定时器功能与 Timer1 相同。QEA 引脚用作定时器时钟输入。

当配置为定时器时， POSCNT 寄存器作为定时器计数寄存器，而 MAXCNT 寄存器用作周期寄存器。当定时器 / 周期寄存器匹配时， QEI 中断标志将被置 1。

此定时器和通用定时器之间惟一的差异是增加了外部递增 / 递减输入选择功能。当 UPDN 引脚为高电平时，定时器将进行递增计数。当 UPDN 引脚为低电平时，定时器将进行递减计数。

注：	工作模式的改变 （即，从 QEI 改变到定时器模式或反之）将不会对定时器 / 位置计数寄存器内容造成影响。

UPDN 控制 / 状态位（QEICON<11>）可用来选择定时器寄存器的计数方向状态。当 UPDN = 1 时，定时器将进行递增计数。当 UPDN = 0 时，定时器将进行递减计数。

此外，控制位 UPDN_SRC （QEICON<0>）用来确定定时器计数方向状态是取决于写入UPDN控制 / 状态位的逻辑状态 （QEICON<11>）还是 QEB 引脚的状态。当 UPDN_SRC = 1 时，定时器计数方向将由 QEB 引脚控制。同样，当 UPDN_SRC = 0 时，定时器计数方向由 UPDN 位确定。

注：	此定时器不支持外部异步计数器工作模式。如果使用外部时钟源，时钟将自动与内部指令周期同步。

13.6 CPU 休眠模式下的 QEI 模块操作

13.6.1 CPU 休眠模式下的 QEI 操作

在 CPU 休眠模式下 QEI 模块将停止工作。

13.6.2 CPU 休眠模式下的定时器操作

在 CPU 休眠模式下，定时器将不工作，因为内部时钟被禁止了。

13.7 CPU 空闲模式下的 QEI 模块操作

由于 QEI 模块可作为正交编码器接口，或 16 位定时器，以下各节介绍了这两种模式下的 QEI 模块操作。

13.7.1 CPU 空闲模式下的 QEI 操作

当 CPU 处于空闲模式时，如果 QEISIDL 位（QEICON<13>）= 0，则 QEI 模块将工作。在 POR 和 BOR 时，该位默认为逻辑 ‘0’。在空闲模式下若要使 QEI 模块停止工作，应将 QEISIDL 位设定为 ‘1’。

13.7.2 CPU 空闲模式下的定时器操作

当 CPU 处于空闲模式且 QEI 模块配置为 16 位定时器模式时，如果 QEISIDL 位（QEICON<13>）= 0，则 16 位定时器将工作。在 POR 和 BOR 时，该位默认为逻辑 ‘0’。在空闲模式下若要使定时器模块停止工作，应将 QEISIDL 位设定为 ‘1’。

如果 QEISIDL 位被清零，定时器将正常工作，就象 CPU 没有进入空闲模式一样。

13.8 正交编码器接口中断

正交编码器接口能够在发生以下事件时产生中断：

- 当 16 位递增 / 递减位置计数器发生计满溢出 / 下溢时产生中断
- 检测到合格的索引脉冲时，或 CNTERR 位置 1 时
- 定时器周期匹配事件 （溢出 / 下溢）
- 门控累加事件

在发生以上任一事件时，QEI 中断标志位 QEIIF 将被置 1。QEIIF 位必须通过软件方式清零。QEIIF 位于 IFS2 状态寄存器中。

可通过相应的中断允许位 QEIIE 允许中断。QEIIE 位于 IEC2 控制寄存器中。

表 **13-1： QEI** 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
QEICON	0122	CNTERR	—	QEISIDL	INDX	UPDN	QEIM2	QEIM1	QEIM0	SWPAB	—	TQGATE	TQCKPS1	TQCKPS0	POSRES	TQCS	UPDN_SRC	0000 0000 0000 0000
DFLTCON	0124	—	—	—	—	—	IMV1	IMV0	CEID	QEOUT	QECK2	QECK1	QECK0	—	—	—	—	0000 0000 0000 0000
POSCNT	0126	位置计数器 <15:0>																0000 0000 0000 0000
MAXCNT	0128	最大计数值 <15:0>																1111 1111 1111 1111

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

14.0 电机控制 PWM 模块

> **注：**本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

此模块简化了产生多种同步脉宽调制输出（Pulse Width Modulated，PWM）的任务。特别是它还能支持以下电源和电机控制应用：

- 三相交流感应电机
- 开关磁阻电机
- 直流无刷（Brushless DC，BLDC）电机
- 不间断电源（Uninterruptible Power Supply，UPS）

PWM 模块具有以下特性：

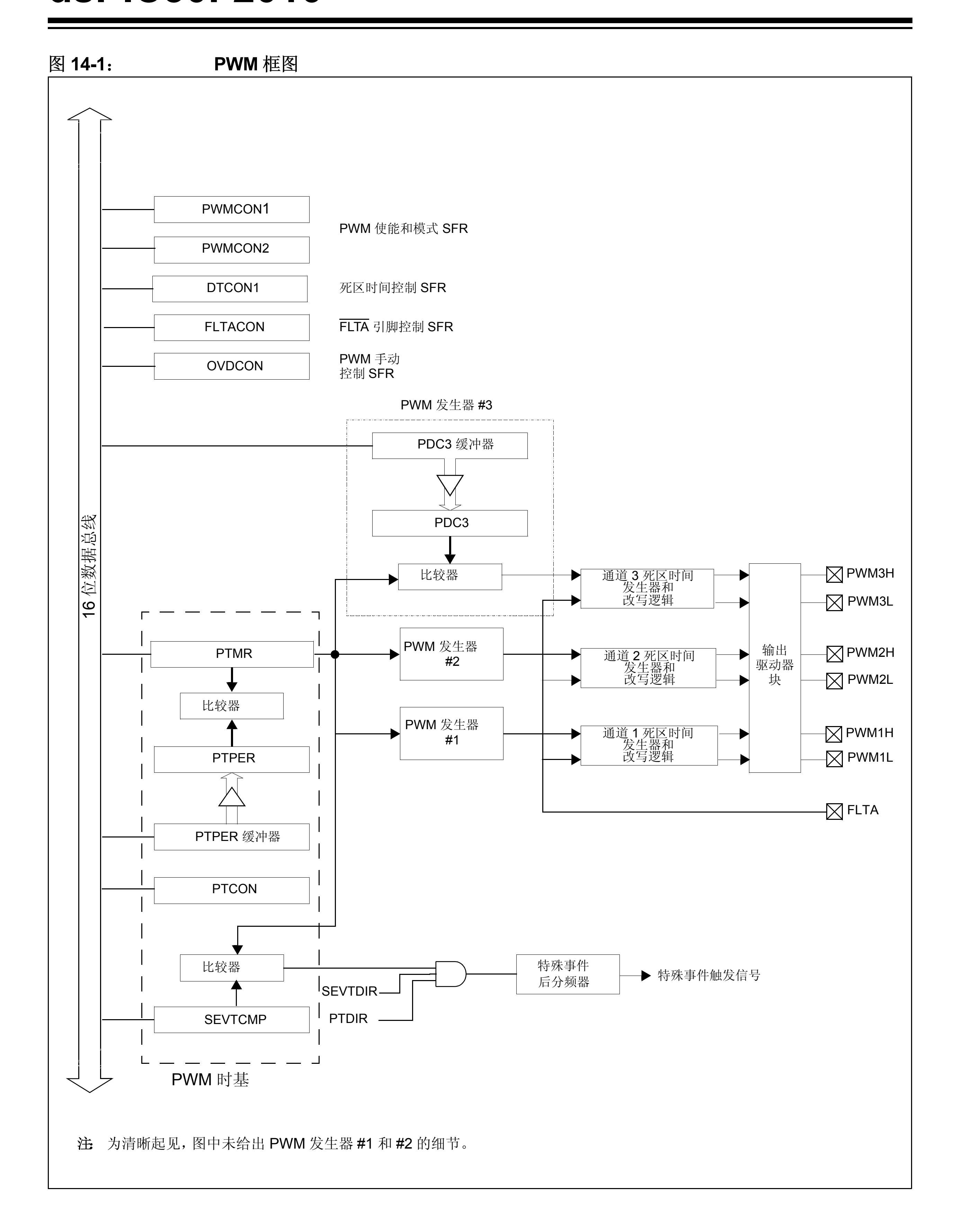
- 6 个具备 3 占空比发生器的 PWM I/O 引脚
- 16 位精度
- 实时更改 PWM 频率
- 边沿对齐和中心对齐输出模式
- 单脉冲产生模式
- 对中心对齐模式下不对称更新的中断支持
- 用于电子换相电机（Electrically Commutative Motor，ECM）控制的输出改写控制
- 用于调度其他外设事件的“特殊事件”比较器
- 可选择将任何一个 PWM 输出引脚驱动至确定状态的 $\overline{\text{FLTA}}$ 引脚

此模块包含 3 个占空比发生器，分别编号为 1 至 3。此模块具有 6 个 PWM 输出引脚，分别编号为 PWM1H/PWM1L 至 PWM3H/PWM3L。6 个 I/O 引脚可组合为 3 个高 / 低端引脚对，分别以下标 H 或 L 表示。对于互补的负载，低端 PWM 引脚的状态总是与高端 I/O 引脚的状态相反。

图 14-1 中显示了电机控制 PWM 模块的简化框图。

PWM 模块可以实现多种工作模式，有利于实现特定的功率控制应用。

图 14-1： PWM 框图



注 为清晰起见，图中未给出 PWM 发生器 #1 和 #2 的细节。

14.1 PWM 时基

PWM 时基由带预分频器和后分频器的 15 位定时器提供。通过 PTMR SFR 可对该时基进行访问。PTMR<15> 为只读状态位，用于表明 PWM 时基的当前计数方向。如果 PTDIR 被清零，表明 PTMR 正进行递增计数。如果 PTDIR 置 1，表明 PTMR 正进行递减计数。可通过 PTCON SFR 对 PWM 时基进行配置。可通过置 1 / 清零 PTCON SFR 中的 PTEN 位使能 / 禁止时基。当用软件清零 PTEN 位时， PTMR 将不会被清零。

PTPER SFR 设定 PTMR 的计数周期。用户必须写入 15 位值至 PTPER<14:0>。当 PTMR<14:0> 中的值与 PTPER<14:0> 中的值相匹配时，时基将会被复位为 0 或在下一个时钟周期反转计数方向，这取决于时基的工作模式。

注：	如果周期寄存器设定为 0x0000，定时器将停止计数，且即使特殊事件值也是 0x0000，也不会产生中断和特殊事件触发信号。该模块将不会更新周期寄存器，即使它已是 0x0000；因此，用户必须禁止该模块以更新周期寄存器。

PWM 时基可配置为 4 种不同的工作模式：

- 自由运行模式
- 单事件模式
- 连续递增 / 递减计数模式
- 带双更新中断的连续递增 / 递减计数模式

通过 PTCON SFR 中的 PTMOD<1:0> 位可选择这 4 种模式。递增 / 递减计数模式支持产生中心对齐 PWM。单事件模式使得 PWM 模块可支持某些电子换相电机（ECM）的脉冲控制。

PWM 时基产生的中断信号取决于模式选择位（PTMOD<1:0>）和 PTCON SFR 中的后分频比控制位（PTOPS<3:0>）。

14.1.1 自由运行模式

在自由运行模式下， PWM 时基进行递增计数直至与时基周期寄存器 （Time Base Period register， PTPER）中的设定值相匹配。PTMR 寄存器将在下一输入时钟边沿复位且只要 PTEN 位保持置 1 时基将继续进行递增计数。

当PWM时基处于自由运行模式时（PTMOD<1:0> = 00），每当计数值与 PTPER 寄存器匹配都将产生一个中断事件且 PTMR 寄存器将被复位为零。在此模式下，可使用后分频比选择位来降低中断事件的频率。

14.1.2 单事件模式

在单事件计数模式下，当 PTEN 位置 1 时 PWM 时基将开始递增计数。当 PTMR 寄存器中的值与 PTPER 寄存器匹配时，PTMR 寄存器将在下一个输入时钟边沿被复位，且 PTEN 位将被硬件清零以中止时基的操作。

当 PWM 时基处于单事件模式 （PTMOD<1:0> = 01）时，当发生与 PTPER 寄存器匹配时将产生一个中断事件，PTMR 寄存器会在下一个输入时钟边沿复位为零且 PTEN 位将被清零。在该模式下，后分频比选择位将不起作用。

14.1.3 连续递增 / 递减计数模式

在连续递增 / 递减计数模式下， PWM 时基将进行递增计数直至与 PTPER 寄存器值相匹配。定时器将在下一个输入时钟边沿开始递减计数。 PTCON SFR 中的 PTDIR 位为只读位，用于表明计数方向。当定时器进行递减计数时， PTDIR 位将被置 1。

在递增 / 递减计数模式 （PTMOD<1:0> = 10）下，每当 PTMR 寄存器值为零时将产生中断事件且 PWM 时基将开始递增计数。在此模式下可使用后分频比选择位来降低中断事件的频率。

14.1.4 双更新模式

在双更新模式（PTMOD<1:0> = 110）下，每当 PTMR 寄存器等于零或出现周期匹配时将产生中断事件。在此模式下，后分频比选择位将不起作用。

双更新模式向用户提供了另外两种功能。首先，控制环带宽可增加一倍，因为在每一个周期内 PWM 占空比可被更新两次。其次，可产生不对称的中心对齐 PWM 波形，可在某些电机控制应用中用于减少输出波形畸变。

注：	将周期寄存器编程设定为 0x0001 可产生一个连续的中断脉冲，因此必须加以避免。

14.1.5 PWM 时基预分频器

PTMR 的输入时钟 （Fosc/4）可通过 PTCON SFR 寄存器中的 PTCKPS<1:0> 控制位选择 1:1、1:4、1:16 或 1:64 预分频比。当发生以下事件时，预分频器计数器将被清零：

- 写入 PTMR 寄存器
- 写入 PTCON 寄存器
- 任何器件复位

当写入 PTCON 时， PTMR 寄存器将不会被清零。

14.1.6 PWM 时基后分频器

PTMR 的匹配输出可选择通过 4 位后分频器进行后分频（可实现 1:1 至 1:16 分频）。

当发生以下事件时，后分频器计数器将被清零：

- 写入 PTMR 寄存器
- 写入 PTCON 寄存器
- 任何器件复位

当写入 PTCON 作时， PTMR 寄存器将不会被清零。

14.2 PWM 周期

PTPER 为 15 位寄存器，用来设定 PWM 时基的计数周期。 PTPER 为双缓冲寄存器。在下列情况下， PTPER 缓冲器内容将被装入 PTPER 寄存器：

- 自由运行和单事件模式：与 PTPER 寄存器匹配后当 PTMR 寄存器复位为零时。
- 递增 / 递减计数模式：当 PTMR 寄存器为零时。

当 PWM 时基被禁止时 （PTEN = 0）， PTPER 缓冲器中的值将被自动装入 PTPER 寄存器。

可使用公式 14-1 确定 PWM 周期。

公式 14-1： PWM 周期

$$T_{PWM} = \frac{T_{CY} \bullet (PTPER + 1)}{(PTMR\ 预分频比)}$$

如果 PWM 时基配置为递增 / 递减计数模式之一，则 PWM 周期可由公式 14-2 确定。

公式 14-2： PWM 周期 （递增 / 递减计数模式）

$$T_{PWM} = \frac{T_{CY} \bullet 2 \bullet (PTPER + 0.75)}{(PTMR\ 预分频比)}$$

对于给定器件振荡器和 PWM 频率，最大分辨率（以位为单位）可使用公式 14-3 进行确定：

公式 14-3： PWM 分辨率

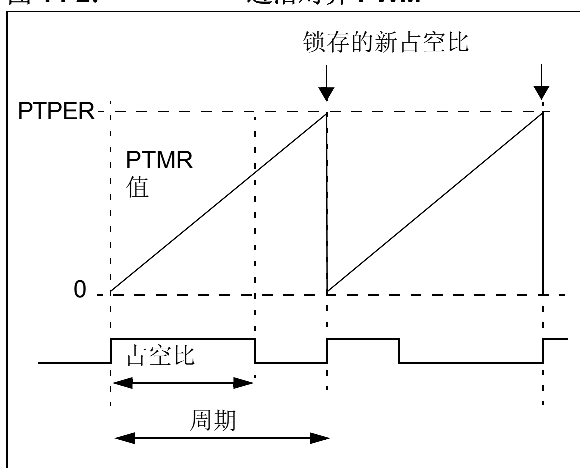
$$分辨率 = \frac{\log (2 \bullet T_{PWM} / T_{CY})}{\log (2)}$$

14.3 边沿对齐 PWM

当 PWM 时基处于自由运行模式或单事件模式时，模块将产生边沿对齐 PWM 信号。对于边沿对齐 PWM 输出，输出信号的周期由 PTPER 中的值指定，而占空比由相应的占空比寄存器指定（见图 14-2）。PWM 输出将在周期的起始 （PTMR = 0）驱动为有效状态，而当占空比寄存器中的值与 PTMR 匹配时被驱动为无效状态。

如果特定占空比寄存器的值为零，则对应 PWM 引脚上的输出会在整个 PWM 周期中处于无效状态。此外，如果占空比寄存器中的值大于 PTPER 寄存器中的值，则 PWM 引脚的输出将会在整个 PWM 周期中保持有效状态。

图 14-2： 边沿对齐 PWM



14.4 中心对齐 PWM

当 PWM 时基配置为递增 / 递减计数模式时（见图 14-3），模块将产生中心对齐的 PWM 信号。

当占空比寄存器值与 PTMR 的值相匹配且 PWM 时基进行递减计数 （PTDIR = 1）时， PWM 比较输出将被驱动为有效状态。当 PWM 时基进行递增计数 （PTDIR = 0）且 PTMR 寄存器中的值与占空比值匹配时， PWM 比较输出将被驱动为无效状态。

如果特定占空比寄存器中的值为零，则整个 PWM 周期内对应 PWM 引脚上的输出将为无效状态。此外，如果占空比寄存器中的值等于 PTPER 寄存器中的值，则整个 PWM 周期内 PWM 引脚上的输出将为有效状态。

图 14-3： 中心对齐 PWM

周期 /2
PTPER
PTMR
值
占空比
0
周期

14.5 PWM 占空比比较单元

有 4 个 16 位特殊功能寄存器（PDC1、PDC2、PDC3 和 PDC4）用来指定 PWM 模块的占空比值。

每一个占空比寄存器中的值负责确定 PWM 输出处于有效状态的时间。占空比寄存器为 16 位宽。占空比寄存器的 LSb 用来确定开始时是否出现 PWM 边沿。因此，PWM 的分辨率实际上提高了一倍。

14.5.1 占空比寄存器缓冲器

4 个 PWM 占空比寄存器采用双缓冲方式，因此可使 PWM 输出更新时不会产生毛刺。对于每一个占空比，有一个用户可访问的占空比寄存器，而另外一个占空比寄存器保存当前 PWM 周期使用的实际比较值。

对于边沿对齐 PWM 输出，当计数值与 PTPER 寄存器值匹配且 PTMR 寄存器被复位时，将更新占空比值。当 PWM 时基被禁止 （PTEN = 0）且 PWMCON2 寄存器中的 UDIS 位被清零时，占空比缓冲器的内容将被自动装入占空比寄存器。

如果 PWM 时基处于递增 / 递减计数模式，则当 PTMR 寄存器值为零且 PWM 时基开始递增计数时将采用新的对占空比值。当 PWM 时基被禁止（PTEN = 0）时，占空比缓冲器中的值将被自动装入占空比寄存器。

如果PWM时基处于双更新功能的递增/递减计数模式，当PTMR 寄存器值为零且PTMR 的值与 PTPER 寄存器的值匹配时，将采用新的占空比值。当 PWM 时基被禁止（PTEN = 0），占空比缓冲器中的值将被自动装入占空比寄存器。

14.6 互补 PWM 操作

在互补工作模式下，每对 PWM 输出都是互补的 PWM 信号。对于一段两个引脚的输出均为无效的短暂时间，可选择在器件开关过程中插入一个死区时间 （参见**第 14.7 节 "死区时间发生器"**）。

在互补模式下，占空比比较单元将按如下分配给相应的 PWM 输出：

- PDC1 寄存器控制 PWM1H/PWM1L 输出
- PDC2 寄存器控制 PWM2H/PWM2L 输出
- PDC3 寄存器控制 PWM3H/PWM3L 输出

将PWMCON1 SFR 中相应的 PMODx 位清零可使 PWM I/O 引脚对工作于互补模式。在器件复位时， PWM I/O 引脚将被设定为互补模式。

14.7 死区时间发生器

当任一 PWM I/O 引脚对工作于互补输出模式时，可使用死区时间发生功能。 PWM 输出采用推挽驱动电路。由于功率输出器件的导通和关断都需要时间，因此在一对互补 PWM 输出中一个开关器件的关断和另一个开关器件的导通之间必须间隔一定的时间。

14.7.1 死区时间发生器

PWM 模块的每一对互补输出都有一个 6 位的递减计数器，用于插入死区时间。如图 14-4 中所示，每个死区时间单元都有与占空比比较输出相连的上升沿和下降沿检测器。

14.7.2 死区时间范围

通过指定输入时钟预分频值和一个 6 位无符号值可设定死区时间单元提供的死区时间。

死区时间单元提供了 4 种输入时钟预分频比选项，用户可根据器件的工作频率选择适当的死区时间范围。使用 DTCON1 SFR 中的 DTAPS<1:0> 和 DTBPS<1:0> 控制位可选择死区时间时钟预分频值。死区时间值可使用 4 种时钟预分频比选择之一（TCY、2TCY、4TCY或8TCY）。

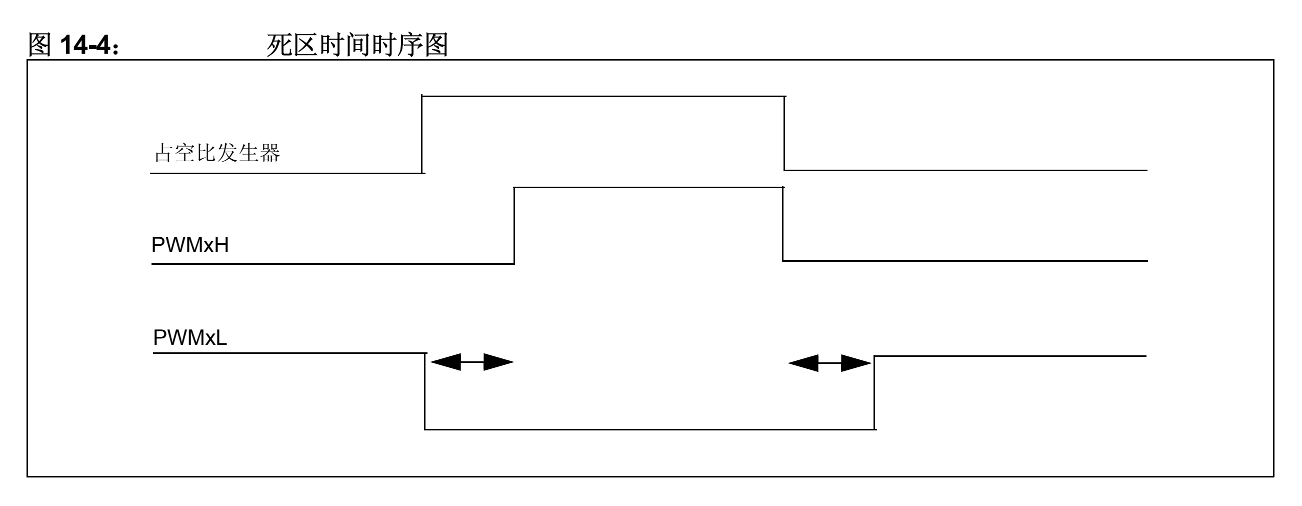
在选择预分频值之后，通过装入 6 位无符号值至 DTCON1 SFR 可对死区时间进行调整。

发生以下事件时，死区时间单元预分频器将被清零：

- 由于占空比比较边沿事件导致装入递减定时器。
- 写入 DTCON1 寄存器。
- 任何器件复位。

注：	当 PWM 模块处于工作状态时（PTEN = 1），用户不应对 DTCON1 值进行修改，否则可能导致意外结果的发生。

图 14-4：　　死区时间时序图



14.8 独立的 PWM 输出

在驱动某些类型的负载时，需要采用独立的PWM输出。当 PWMCON1 寄存器中相应的 PMOD 位置 1 时，则相应的 PWM 输出对将工作于独立输出模式。当模块工作于独立模式时，相邻 PWM I/O 引脚之间将不会实现死区时间控制而 I/O 引脚对将允许同时处于有效状态。

在独立模式下，每一个占空比发生器将被同时连接到 PWM 输出对中的两个 I/O 引脚。通过相关的占空比寄存器和 OVDCON 寄存器中的相应位，用户可为工作于独立模式的每一个 PWM I/O 引脚选择以下的信号输出选项：

- I/O 引脚输出 PWM 信号
- I/O 引脚处于无效状态
- I/O 引脚处于有效状态

14.9 单脉冲 PWM 操作

当 PTCON 控制位 PTMOD<1:0> = 10 时， PWM 模块将产生单脉冲输出。在单脉冲模式下仅可产生边沿对齐输出。在单脉冲模式下，当 PTEN 位置 1 时， PWM I/O 引脚将被驱动为有效状态。当计数值与占空比寄存器设定值匹配时，PWM I/O 引脚将被驱动为无效状态。当计数值与 PTPER 寄存器匹配时， PTMR 寄存器将被清零，所有处于有效状态的 PWM I/O 引脚将被驱动为无效状态， PTEN 位将被清零且将产生中断。

14.10 PWM 输出改写

通过 PWM 输出改写控制位 , 用户可手动驱动 PWM I/O 引脚至指定逻辑状态，而不受占空比比较单元的影响。

OVDCON 寄存器中包含了所有与 PWM 输出改写功能相关的控制位。 OVDCON 寄存器的高字节包含 6 个数据位，即 POVDxH<3:1> 和 POVDxL<3:1>，用于确定哪些 PWM I/O 引脚输出将被改写。OVDCON 寄存器的低字节包含 POUTxH<3:1> 和 POUTxL<3:1> 6 个数据位，用于确定当通过 POVD 位对特定输出进行改写时的 PWM I/O 引脚的状态。

14.10.1 互补输出模式

当通过 OVDCON 寄存器驱动 PWMxL 引脚为有效状态时，输出信号将被强制为与输出对中对应 PWMxH 引脚输出状态相反的状态。当通过手动方式对 PWM 通道进行改写时，死区时间插入仍然有效。

14.10.2 改写同步

如果 PWMCON2 寄存器中的 OSYNC 位置 1，所用通过 OVDCON 寄存器执行的输出改写操作将与 PWM 时基同步。同步输出改写将发生在以下时刻：

- 若是边沿对齐模式，则当 PTMR 为零时。
- 若是中心对齐模式，则当 PTMR 为零或当 PTMR 值与 PTPER 匹配时。

14.11 PWM 输出和极性控制

有 3 个与 PWM 模块有关的器件配置位用来提供 PWM 输出引脚控制：

- HPOL 配置位
- LPOL 配置位
- PWMPIN 配置位

FPORBOR 配置寄存器中的这 3 个位 （见第 21 章）将与位于 PWMCON1 SFR 中的 3 个 PWM 使能位（PWMEN<3:1>）配合工作。这些控制位和 PWM 使能位确保在器件发生复位后 PWM 引脚能够处于正确的状态。配置位 PWMPIN 使得 PWM 模块输出可选择在器件复位时被使能。如果 PWMPIN = 0，在复位时 PWM 输出将被驱动至无效状态。如果 PWMPIN = 1（默认），PWM 模块将为三态。 HPOL 位用于指定 PWMxH 输出的极性，而 LPOL 位指定 PWMxL 输出的极性。

14.11.1 输出引脚控制

PWMCON1 SFR 中的 PEN<3:1>H 和 PEN<3:1>L 控制位将分别用于使能每一个高端 PWM 输出引脚和每一个低端 PWM 输出引脚。如果某个 PWM 输出引脚未被使能，则该引脚将被视作一般 I/O 引脚。

14.12 PWM $\overline{FLTA}$ 引脚

有一个与 PWM 模块相关的故障引脚，即 $\overline{FLTA}$ 引脚（FLTA）。当使能时，可以选择用该引脚将每个 PWM I/O 引脚驱动至定义的状态。

14.12.1 故障引脚使能位

FLTACON SFR 具有 4 个控制位，用于确定特定 PWM I/O引脚对是否由$\overline{FLTA}$输入引脚控制。欲使能某一PWM I/O 引脚对的 $\overline{FLTA}$ 改写功能，必须将 FLTACON 寄存器中的相应位置 1。

如果寄存器 FLTACON 中所有的使能位都被清零，则 $\overline{FLTA}$ 输入引脚对 PWM 模块将没有影响，且该引脚可用作一般的中断引脚或 I/O 引脚。

注：	$\overline{FLTA}$ 引脚逻辑可独立于 PWM 逻辑进行工作。如果 FLTACON 寄存器中的所有使能位被清零，则 $\overline{FLTA}$ 引脚可用作一般中断引脚。每一个 FLTA 引脚具有一个中断向量，中断标志位和与之相关的中断优先级位。

14.12.2 故障状态

FLTACON 特殊功能寄存器具有 8 个位，用于确定每一个 PWM I/O 引脚被 $\overline{FLTA}$ 输入改写时的状态。当这些位清零时，PWM I/O 引脚将被驱动至无效状态。如果这些位置 1，PWM I/O 引脚将被驱动至有效状态。有效和无效状态与为每个 PWM I/O 引脚所定义的极性有关（HPOL 和 LPOL 极性控制位）。

14.12.3 故障输入模式

$\overline{FLTA}$ 输入引脚具有两种工作模式：

- **锁存模式：**当 $\overline{FLTA}$ 引脚驱动为低电平时，PWM 输出将进入 FLTACON 寄存器所定义的状态。 PWM 输出将保持在此状态，直到 $\overline{FLTA}$ 引脚被驱动为高电平并且相应的中断标志由软件清零。当这两种情况都发生时， PWM 输出将在下一个 PWM 周期开始时或在半周期边界返回到正常工作状态。如果中断标志在 $\overline{FLTA}$ 状态结束前清零， PWM 模块将等到 $\overline{FLTA}$ 引脚不再有效时才恢复输出。
- **逐周期模式：**当 $\overline{FLTA}$ 输入引脚被驱动为低电平时，只要 $\overline{FLTA}$ 引脚保持为低电平， PWM 输出将会一直保持在定义的 $\overline{FLTA}$ 状态。在 $\overline{FLTA}$ 引脚被驱动为高电平后， PWM 输出将在下一个 PWM 周期开始时或半周期边界返回正常状态。

$\overline{FLTA}$输入引脚的工作模式可通过FLTACON特殊功能寄存器中的 FLTAM 控制位进行选择。

$\overline{FLTA}$ 引脚可通过软件方式进行手动控制。

14.13 PWM 更新锁定

对于某些复杂的 PWM 应用，用户可能需要在某一时刻对多达 4 个占空比寄存器以及时基周期寄存器 PTPER 进行写入操作。在某些应用中，在装入新的占空比和周期设定值供模块使用之前写入所有的缓冲寄存器是很重要的。

通过将 PWMCON2 SFR 中的 UDIS 控制位置 1 可使能 PWM 更新锁定功能。UDIS 位会影响所有占空比缓冲寄存器和PWM时基周期缓冲器PTPER。当UDIS = 1时，占空比或周期值将不会发生变化。

14.14 PWM 特殊事件触发器

PWM 模块具有一个特殊事件触发器，可使 A/D 转换与 PWM 时基同步。通过对 A/D 采样和转换的时间进行编程设定，可使其发生在 PWM 周期中的任何时刻。特殊事件触发器使用户将采集A/D 转换结果的时间和更新占空比值的时间之间的延迟减到最小。

PWM 特殊事件触发器使用一个称为 SEVTCMP 的 SFR 以及 5 个控制位对其操作进行控制。用于产生特殊事件触发信号的 PTMR 值装入 SEVTCMP 寄存器。当 PWM 时基处于递增 / 递减计数模式时，还需要一个控制位指定特殊事件触发信号的计数方向。此计数方向是通过使用 SEVTCMP SFR 中的 SEVTDIR 控制位选择的。如果 SEVTDIR 位清零，特殊事件触发信号将在 PWM 时基的递增计数周期产生。如果 SEVTDIR 位置 1，特殊事件触发信号将在 PWM 时基的递减计数周期产生。如果 PWM 时基不配置为递增 / 递减计数模式，SEVTDIR 控制位不起作用。

14.14.1 特殊事件触发器后分频器

PWM特殊事件触发器有一个允许后分频比为1:1 到1:16 的后分频器。通过写入 PWMCON2 SFR 中的 SEVOPS<3:0> 控制位可配置后分频器。

特殊事件输出后分频器在发生下列事件时清零：

- 任何写入 SEVTCMP 寄存器的操作
- 任何器件复位

14.15 CPU 休眠模式下的 PWM 操作

$\overline{\text{FLTA}}$ A 和 $\overline{\text{FLTA}}$ B 输入引脚能将 CPU 从休眠状态唤醒。在休眠模式下，如果任何一个 FLTA 引脚被驱动为低电平， PWM 模块将产生中断。

14.16 CPU 空闲模式下的 PWM 操作

PTCON SFR 包含 PTSIDL 控制位。该位用于确定当器件进入空闲模式时 PWM 模块将继续工作还是停止工作。如果 PTSIDL = 0，空闲模式下模块将继续工作。如果 PTSIDL = 1，只要 CPU 仍然处于空闲模式，模块将停止工作。

表 14-1： **PWM 寄存器映射**

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
PTCON	01C0	PTEN	—	PTSIDL	—	—	—	—	—	PTOPS<3:0>				PTCKPS<1:0>		PTMOD<1:0>		0000 0000 0000 0000
PTMR	01C2	PTDIR	PWM 定时器计数值															0000 0000 0000 0000
PTPER	01C4	—	PWM 时基周期寄存器															0011 1111 1111 1111
SEVTCMP	01C6	SEVTDIR	PWM 特殊事件比较寄存器															0000 0000 0000 0000
PWMCON1	01C8	—	—	—	—	—	PTMOD3	PTMOD2	PTMOD1	—	PEN3H	PEN2H	PEN1H	—	PEN3L	PEN2L	PEN1L	0000 0000 0111 0111
PWMCON2	01CA	—	—	—	—	SEVOPS<3:0>				—	—	—	—	—	—	OSYNC	UDIS	0000 0000 0000 0000
DTCON1	01CC	DTBPS<1:0>		DTB<5:0>						DTAPS<1:0>		死区时间 A 值						0000 0000 0000 0000
FLTACON	01D0	—	—	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L	FLTAM	—	—	—	—	FAEN3	FAEN2	FAEN1	0000 0000 0000 0000
OVDCON	01D4	—	—	POVD3H	POVD3L	POVD2H	POVD2L	POVD1H	POVD1L	—	—	POUT3H	POUT3L	POUT2H	POUT2L	POUT1H	POUT1L	0011 1111 0000 0000
PDC1	01D6	PWM 占空比 #1 寄存器																0000 0000 0000 0000
PDC2	01D8	PWM 占空比 #2 寄存器																0000 0000 0000 0000
PDC3	01DA	PWM 占空比 #3 寄存器																0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

15.0 SPI 模块

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

串行外设接口（Serial Peripheral Interface，SPI）模块是同步串行接口，可用于与其他外设（如 EEPROM、移位寄存器、显示驱动器和 A/D 转换器等）或者单片机进行通信。SPI 模块与 Motorola 的 SPI 和 SIOP 接口兼容。

15.1 功能说明

每个 SPI 模块由一个 16 位移位寄存器（SPIxSR，其中 x = 1 或 2）和一个缓冲器寄存器（SPIxBUF）组成；移位寄存器用来移入和移出数据。一个控制寄存器（SPIxCON）用来配置模块。此外，一个状态寄存器（SPIxSTAT）用来表明各种状态条件。

串行接口有 4 个引脚：SDIx（串行数据输入）、SDOx（串行数据输出）、SCKx（移位时钟输入或输出）以及 $\overline{\text{SSx}}$（低电平有效从选择）。

在主模式下工作时，SCK 是时钟输出，但在从模式下它是时钟输入。

一组 8 或 16 个时钟脉冲将数据位从 SPIxSR 移出到 SDOx 引脚，同时将 SDIx 引脚的数据位移入 SPIxSR。当传输完成后将产生中断，相应的中断标志位（SPI1IF 或 SPI2IF）置 1。通过中断允许位（SPI1IE 或 SPI2IE）可以禁止该中断。

接收操作是双缓冲的。当接收完字节时，将字节从 SPIxSR 送到 SPIxBUF。

当从 SPIxSR 传输新数据到 SPIxBUF 时，如果接收缓冲器满，模块将置 1 SPIROV 位，表明产生溢出条件。数据从 SPIxSR 向 SPIxBUF 的传送不会完成，新数据将丢失。当 SPIROV 为 1 时，模块将不会对 SCK 电平跳变进行响应；实际上会禁止模块，直到用户软件读 SPIxBUF 为止。

发送写操作同样也是双缓冲的。用户写入 SPIxBUF。当主器件或从器件传输完成后，移位寄存器（SPIxSR）的内容将送至接收缓冲器。如果已经向缓冲器寄存器写了任何发送数据，发送缓冲器的内容将送至 SPIxSR。于是，接收到的数据存放在 SPIxBUF 中，而 SPIxSR 中的发送数据已经准备就绪，可进行下一次传输。

> 注：发送缓冲器（SPIxTXB）和接收缓冲器（SPIxRXB）都映射到相同的寄存器地址 SPIxBUF。

在主模式下，通过对系统时钟进行预分频来产生时钟。只要数据写入 SPIxBUF 就会被发送。在最后一个位传送的过程中产生中断。

在从模式下，SCK 上出现外部时钟时，进行数据的发送和接收。同样，当最后一个位被锁存时产生中断。如果使能了 $\overline{\text{SSx}}$ 控制，那么仅当 $\overline{\text{SSx}}$ 为低时使能发送和接收。在 $\overline{\text{SSx}}$ 模式下，当 $\overline{\text{SSx}}$ 为高时 SDOx 输出将被禁止。

提供给模块的时钟是 Fosc/4。而后，使用主预分频因子（PPRE<1:0>）和辅助预分频因子（SPRE<2:0>），对这个时钟进行分频。CKE 位决定发送的时钟边沿，是在时钟从有效状态跳变到空闲状态进行发送，还是反之。CKP 位选择时钟的空闲状态（高电平或低电平）。

15.1.1 字和字节通信

控制位 MODE16（SPIxCON<10>）允许模块在 8 位或 16 位模式下通信。16 位通信，除了发送的位数是 16 而不是 8 以外，与 8 位通信相同。

在更改 MODE16 位之前，用户软件必须禁止模块。用户更改 MODE16 位时，SPI 模块将复位。

8 位和 16 位工作之间的基本区别是，8 位工作模式下数据是从 SPIxSR 的 bit 7 开始发送的；而在 16 位工作模式下，则是从 SPIxSR 的 bit 15 开始发送的。在两种模式下，数据都会移入 SPIxSR 的 bit 0。

15.1.2 SDOx 禁止

SPIxCON 寄存器中的控制位 DISSDO，用来禁止 SDOx 输出。这样可以使 SPIx 模块配置为仅输入的工作模式。SDO 也可用作一般 I/O 口。

15.2 帧 SPI 支持

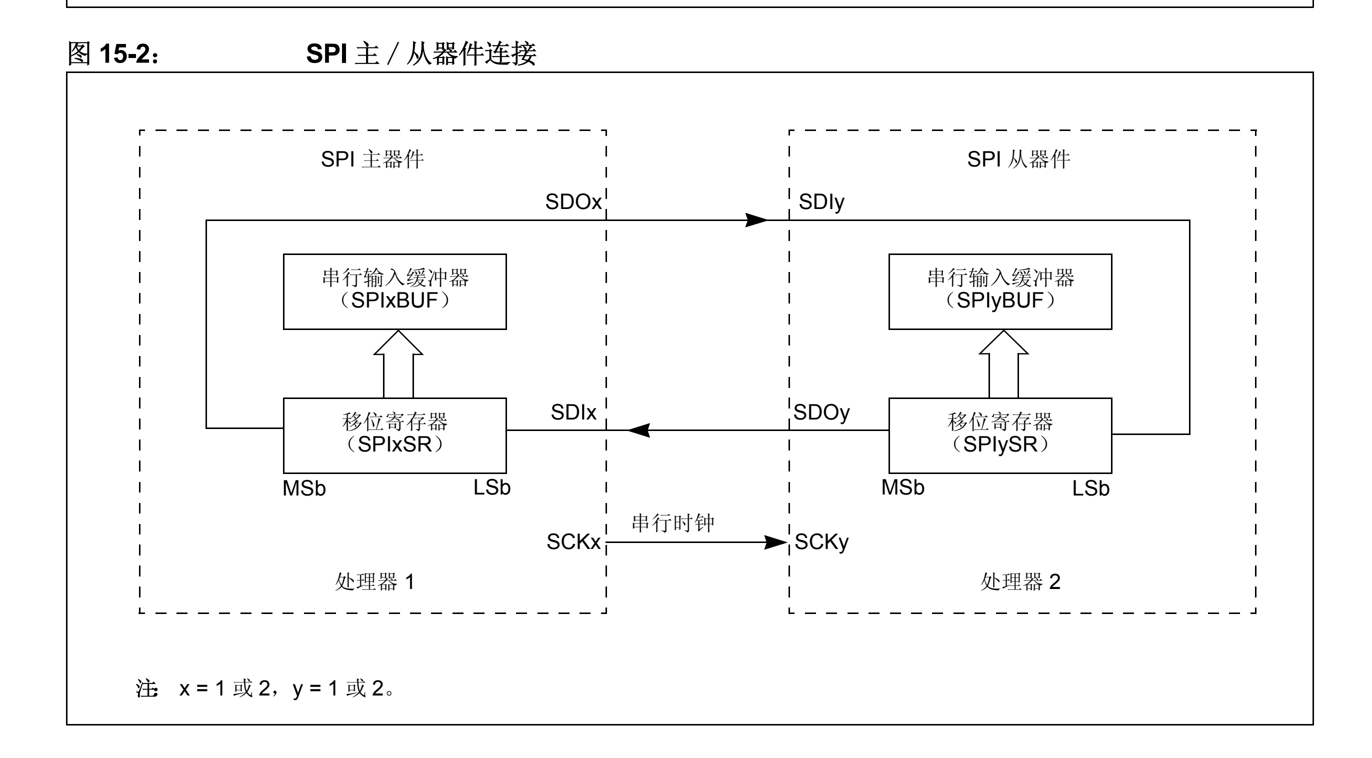
当工作在主模式或从模式时，模块支持基本的帧 SPI 协议。控制位 FRMEN 使能帧 SPI 模式支持，并使 $\overline{\text{SSx}}$ 引脚用作帧同步脉冲（FSYNC）功能。控制位 SPIFSD 决定 $\overline{\text{SSx}}$ 引脚是输入还是输出（即，模块是接收还是产生帧同步脉冲）。帧脉冲在一个 SPI 时钟周期内是高电平有效脉冲。当帧同步使能时，仅在 SPI 时钟的下一个发送边沿开始发送数据。

图 15-1： SPI 框图

内部
数据总线
读
写
SPIxBUF
SPIxBUF
接收
发送
SDIx
SPIxSR
bit 0
SDOx
移位
时钟
SS & FSYNC
控制
时钟
控制
边沿
选择
SSx
辅助
预分频器
1:1–1:8
主预分频器
1:1, 1:4,
1:16, 1:64
FCY
SCKx
使能主时钟

注 x = 1 或 2。

图 15-2： SPI 主 / 从器件连接



15.3 从选择同步

$\overline{\text{SSx}}$ 引脚允许同步从模式。必须使用 $\overline{\text{SSx}}$ 引脚控制使能位（SSEN = 1）把 SPI 配置为 SPI 从模式。当 $\overline{\text{SSx}}$ 引脚为低时，使能发送和接收且驱动 SDOx 引脚。当 $\overline{\text{SSx}}$ 引脚变高时，将不再驱动 SDOx 引脚。另外，可以重新同步 SPI 模块，所有的计数器 / 控制电路都将复位。于是，当 $\overline{\text{SSx}}$ 引脚再次为低时，发送 / 接收又将从 MSb 开始，即使 $\overline{\text{SSx}}$ 是在发送 / 接收过程中变高。

15.4 CPU 休眠模式期间的 SPI 工作

在休眠模式期间，SPI 模块关闭。SPIx 模块正在进行处理时，如果 CPU 进入休眠模式，则发送 / 接收将中止。

在休眠模式下，发送器和接收器将停止。但进入或退出休眠模式不影响寄存器内容。

15.5 CPU 空闲模式期间的 SPI 工作

当器件进入空闲模式时，所有的系统时钟源继续工作。SPISIDL 位（SPIxSTAT<13>）选择 SPI 模块在空闲模式下是停止工作还是继续工作。如果 SPISIDL = 0，模块将继续工作。如果 SPISIDL = 1，模块将停止。

表 **15-1**： **SPI1** 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
SPI1STAT	0220	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI1CON	0222	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI1BUF	0224	发送和接收缓冲器																0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器中各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

16.0 I²C™ 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

I²C 模块（16 位接口）为 I²C 串行通信标准的从模式和多主器件模式提供完全的硬件支持。

模块具有以下主要特性：

- I²C 接口支持主、从模式工作。
- I²C 从模式支持 7 位和 10 位地址。
- I²C 主模式支持 7 位和 10 位地址。
- I²C 端口允许主器件和从器件之间的双向传输。
- I²C 端口的串行时钟同步可以用作握手机制来暂停和继续串行传输 （SCLREL 控制）。
- I²C 支持多主器件工作；检测总线冲突并相应地进行仲裁。

16.1 功能说明

硬件完全实现了 I²C 标准和快速模式规范的全部主从功能，以及 7 位和 10 位寻址。

从而， I²C 模块既可作为 I²C 总线上的主器件工作，也可作为 I²C 总线上的从器件工作。

16.1.1 各种 I²C™ 模式

支持 I²C 模块工作在以下模式：

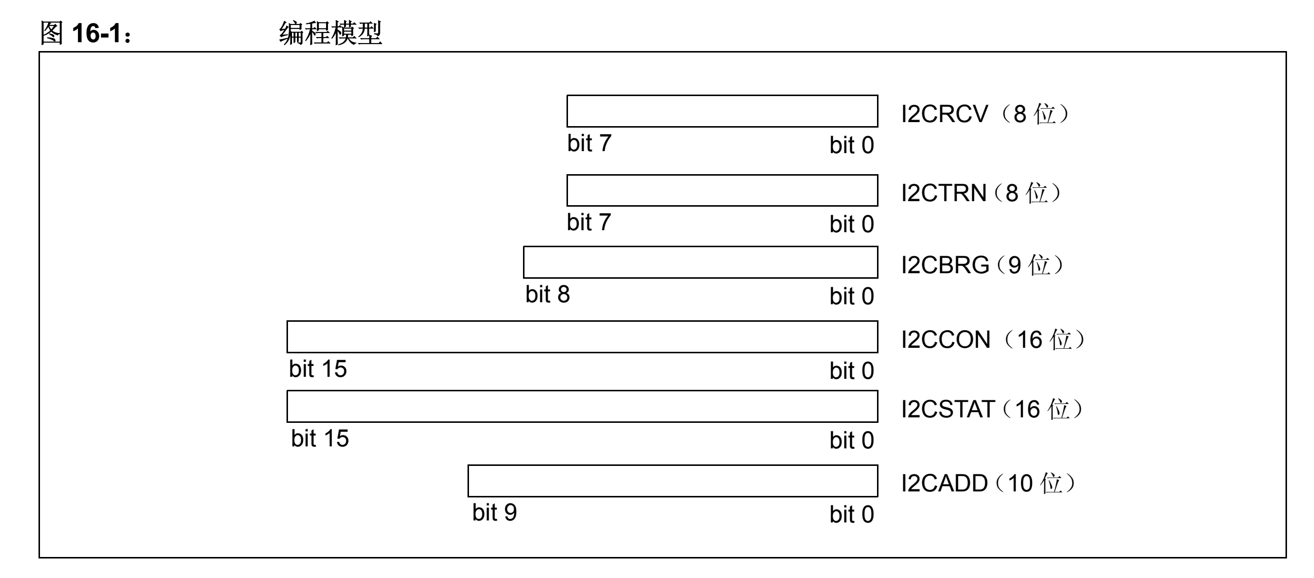
- 7 位地址的 I²C 从模式
- 10 位地址的 I²C 从模式
- 7 位或 10 位地址的 I²C 主模式

参见图 16-1 中的 I²C 编程模型。

16.1.2 I²C™ 模块的引脚配置

I²C 有一个双引脚接口，SCL 引脚是时钟线，SDA 引脚是数据线。

图 **16-1**： 编程模型



16.1.3 I²C™ 寄存器

I2CCON 和 I2CSTAT 分别是控制寄存器和状态寄存器。I2CCON 寄存器是可读且可写的。I2CSTAT 的低 6 位是只读的，它的其余位则是可读且可写的。

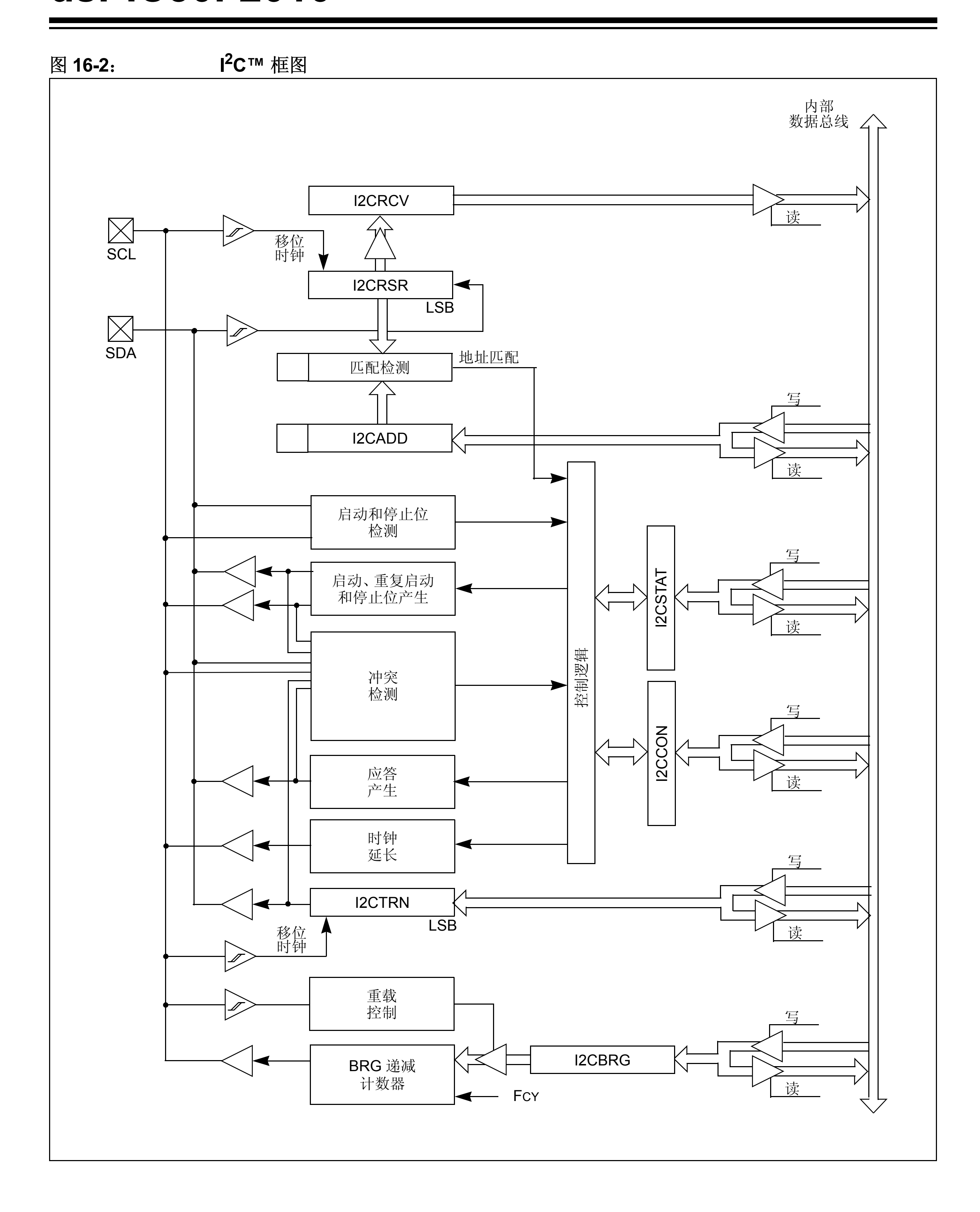
I2CRSR 是移位寄存器，用来对数据进行移位。I2CRCV 是接收缓冲寄存器，可写入、读出数据字节，如图 15-1 所示。 I2CTRN 是发送寄存器，发送操作期间，字节将会写入 I2CTRN，如图 16-2 所示。

I2CADD 寄存器存放从地址。状态位 ADD10 表明是否为 10 位地址模式。 I2CBRG 用来保存波特率发生器（BRG）的重载值。

在进行接收时，I2CRSR 和 I2CRCV 一起形成双缓冲接收器。在 I2CRSR 接收到一个完整的字节后，字节被送至 I2CRCV 并产生中断脉冲。在发送期间， I2CTRN 不是双缓冲的。

注： 在 10 位寻址方式下，在重复启动条件之后，用户只需匹配前 7 位地址。

图 16-2： I2C™ 框图



内部
数据总线
I2CRCV
读
SCL
移位
时钟
I2CRSR
LSB
SDA
匹配检测
地址匹配
写
I2CADD
读
启动和停止位
检测
写
启动、重复启动
和停止位产生
I2CSTAT
读
冲突
检测
控制逻辑
写
I2CCON
应答
产生
读
时钟
延长
写
I2CTRN
LSB
读
移位
时钟
重载
控制
写
BRG 递减
计数器
I2CBRG
读
FCY

16.2 I^{2}C™ 模块地址

I2CADD 寄存器包含从模式地址，它是 10 位宽的寄存器。

如果 A10M 位（I2CCON<10>）为 0，模块把地址解释为7位地址。在接收到地址时，将收到的地址与I2CADD寄存器中的低 7 位进行比较。

如果 A10M 位为 1，将认为地址是 10 位地址。在接收地址时，将收到的地址与二进制数“11110 A9 A8”（其中 A9 和 A8 是 I2CADD 的最高两位）进行比较。如果匹配的话，下一个地址将与 I2CADD 的低 8 位进行比较，这是 10 位寻址协议规定的。

表 16-1： dsPIC30F 支持的 7 位 I^{2}C™ 从地址

0x00	广播呼叫地址或启动字节
0x01-0x03	保留
0x04-0x07	Hs 模式主机码
0x08-0x77	有效 7 位地址
0x78-0x7B	有效 10 位地址 （低 7 位）
0x7C-0x7F	保留

16.3 I^{2}C™ 7 位从模式的工作

一旦使能 （I2CEN = 1），从模块将等待一个启动位出现（即，I^{2}C 模块“空闲”）。检测到启动位之后，8个位将移入I2CRSR，然后地址将与I2CADD进行比较。在 7 位寻址方式下 （A10M = 0），位 I2CADD<6:0> 与位 I2CRSR<7:1> 进行比较，而 I2CRSR<0> 是 R_W位。所有进入的位都是在 SCL 的上升沿采样的。

如果地址匹配的话，将发送一个应答，在第九个位（ACK）的下降沿，从事件中断标志 （SI2CIF）置 1。地址匹配不影响 I2CRCV 缓冲器或 RBF 位的内容。

16.3.1 从模式发送

如果收到的 R_W 位为 1，串行端口将进入发送模式。模块将在第九位发送 ACK，并使 SCL 保持为 0，直到CPU 写 I2CTRN 进行响应为止。通过把 SCLREL 置 1 释放 SCL，而后数据的 8 个位被移出。数据位在 SCL 下降沿移出，因此 SCL 为高 （参见时序图）时 SDA 有效。将在第九个时钟脉冲的下降沿发出中断脉冲，而与从主器件接收到的 ACK 状态无关。

16.3.2 从模式接收

如果地址匹配时，接收到的 R_W 位是 0，就将启动接收模式。在 SCL 上升沿采样进入的位。接收到 8 个位之后，如果 I2CRCV 未满或者 I2COV 没有置 1，I2CRSR中的数据将被送至 I2CRCV。在第九个时钟发送 ACK。

如果 RBF 标志置 1，表明 I2CRCV 仍然存有前一次收到的数据（RBF = 1），则不发送 ACK；然而，仍将产生中断脉冲。如果出现溢出， I2CRSR 的内容不会被装载到 I2CRCV 中。

注：	如果 I2COV 位 为 1 且 RBF 标志为 0，将装入 I2CRCV。在这种情况下，要对I2CRCV进行读操作，但在下一次接收操作发生之前，不会清零 I2COV 位。不发送应答（ACK = 1），但 I2CRCV 位会被更新。

16.4 I^{2}C™ 10 位从模式的工作

在 10 位寻址方式下，基本的接收和发送操作与 7 位寻址方式下的操作相同。不过，地址匹配的判据更加复杂。

I^{2}C规范要求，对于写操作，必须在一个启动位后跟两个地址字节来寻址从器件。

A10M控制位置1表明I2CADD中的地址是10位地址，而不是 7 位地址。报文地址首字节的地址检测协议，对于7位和10位报文地址而言，是完全相同的，但首字节各位进行比较的机制不同。

I2CADD 存放完整的 10 位地址。在接收到启动位后的地址时， I2CRSR <7:3> 与立即数 11110（缺省 10 位地址）进行比较，I2CRSR<2:1> 与 I2CADD<9:8> 进行比较。如果出现匹配且 R_W = 0，则发出中断脉冲。ADD10 位清零，表明地址部分匹配。如果地址不匹配，或者 R_W = 1，则 ADD10 位清零，模块返回到空闲状态。

然后，接收地址的低字节，与 I2CADD<7:0> 进行比较。如果出现匹配，则产生中断脉冲且置 1 ADD10 位，表明 10 位地址完全匹配。如果地址不匹配的话， ADD10位清零，模块返回到空闲状态。

16.4.1 10 位地址从模式接收

一旦以上述方式使用完整的10位地址寻址到从器件（我们把这个状态叫做“PRIOR_ADDR_MATCH”），主器件就可以开始送数据字节，从器件接收。

16.4.2 10 位地址从模式发送

一旦寻址到从器件，主器件就可以产生重复的启动位，复位地址的高字节，并置 1 R_W 位，但不产生停止位，于是开始进行从器件发送操作。

16.5 自动时钟低电平时间延长

当模块工作在从模式下时，通过时钟低电平时间延长，模块能够同步读缓冲器与写主器件。

16.5.1 发送时钟低电平时间延长

在 10 位或 7 位地址发送模式下，如果 TBF 位清零（这表明缓冲器为空），则在第九个时钟的下降沿后清零 SCLREL 位，就实现了时钟延长。

在从发送模式下，始终进行时钟延长，无论 STREN 位的状态为何。

在发送序列的第九个时钟之后，发生时钟同步。如果器件在第九个时钟的下降沿采样到 $\overline{ACK}$，并且 TBF 位仍然清零的话，则 SCLREL 位自动清零。 SCLREL 的清零，将使得 SCL 线变低。在允许继续发送之前，用户中断服务程序必须置 1 SCLREL 位。通过使 SCL 线保持为低，在主器件启动另一个发送序列之前，用户就有时间执行中断服务程序并载入 I2CTRN 的内容。

注 1：	如果用户载入 I2CTRN 的内容，在第九个时钟下降沿之前置 1 TBF 位，则 SCLREL 位将不会被清零，也不会发生时钟延长。
2：	可以在软件中置 1 SCLREL 位，无论 TBF 位的状态为何。

16.5.2 接收时钟低电平时间延长

I2CCON 寄存器中的 STREN 位可以用来使能从件接收模式下的时钟低电平时间延长。当 STREN 位置 1 时，SCL 引脚将在每个数据接收序列结束后保持为低。

16.5.3 7 位寻址时的时钟低电平时间延长（STREN = 1）

从接收模式下，当 STREN 置 1 时，缓冲器已满时 SCL 线保持为低。延长 SCL 输出低电平时间的方法，对于使用 7 位和 10 位寻址的从接收模式是相同的。

在接收序列第九个时钟之后进行时钟低电平时间延长。在 ACK 序列末端的第九个时钟的下降沿，如果 RBF 位置 1 的话， SCLREL 位将自动清零，这将强制 SCL 输出保持为低。在允许继续接收之前，用户中断服务程序必须置 1 SCLREL 位。通过使 SCL 线保持为低，在主器件启动另一个接收序列之前，用户就有时间执行中断服务程序并读出 I2CRCV 的内容。这将防止缓冲器溢出。

注 1：	如果用户读 I2CRCV 的内容，在第九个时钟下降沿之前置 1 RBF 位，则 SCLREL 位将不会被清零，也不会发生时钟延长。
2：	可以在软件中置 1 SCLREL 位，无论 RBF 位的状态为何。为了防止溢出，在下一个接收序列之前，用户应该在中断服务程序中清零 RBF 位。

16.5.4 10 位寻址时的时钟低电平时间延长（STREN = 1）

在寻址过程中自动进行时钟低电平时间延长。由于模块有寄存器存放完整的地址，协议不必等待地址更新。

寻址过程完成后，每个数据接收或发送序列都要进行时钟延长，如前所述。

16.6 软件控制的时钟低电平时间延长（STREN = 1）

当 STREN 位为 1 时，软件可清零 SCLREL 位，从而软件能够控制时钟延长。思路是把写 SCLREL 位的操作与 SCL时钟进行同步。清零SCLREL位不会使SCL输出保持为低，除非模块检测到 SCL 输出的下降沿且采样到 SCL 为低。在 SCL 线采样为低时，如果用户清零 SCLREL 位，则 SCL 输出将保持为低。SCL 输出将保持为低，一直到 SCLREL 位置 1，且 I^{2}C 总线上所有其他器件都已经释放了 SCL。这确保了写 SCLREL 位的操作，不会违背 SCL 的最小高电平时间要求。

如果 STREN 位为 0，软件对 SCLRE 位的写操作将被忽略，不会对 SCLREL 位产生影响。

16.7 中断

I^{2}C 模块产生两个中断标志：MI2CIF（I^{2}C 主中断标志）和 SI2CIF （I^{2}C 从中断标志）。 MI2CIF 中断标志，会在主器件报文事件完成时置1（从而激活中断）。SI2CIF 中断标志，在检测到发往从器件的报文时置 1 （从而激活中断）。

16.8 斜率控制

对于快速模式 （400 kHz）， I^{2}C 标准要求对 SDA 和 SCL 信号进行斜率控制。如果需要的话，用户可通过控制位 DISSLW 来禁止斜率控制。对于 1 MHz 模式，必须禁止斜率控制。

16.9 IPMI 支持

控制位 IPMIEN 允许模块支持智能外设管理接口（IPMI）。当 IPMIEN 位置 1 时，模块将接受并操作所有地址。

16.10 广播呼叫地址支持

广播呼叫地址能寻址所有器件。当使用广播呼叫地址时，理论上所有器件都应该以应答信号作出响应。

广播呼叫地址是 I^{2}C 协议为特定目的保留的八个地址之一。它由全 0 地址位组成，且 R/W = 0。

当广播呼叫使能位 GCEN 置 1 （I2CCON<15> = 1）时，识别广播呼叫地址。在检测到启动位后，将 8 位移入 I2CRSR，并将地址与 I2CADD 进行比较，同时也与广播呼叫地址进行比较，广播呼叫地址固化在硬件中。

如果广播呼叫地址匹配，第八个时钟之后 I2CRSR 被送至 I2CRCV，RBF 标志在第九个位（$\overline{\text{ACK}}$ 位）的下降沿置 1，主事件中断标志 （MI2CIF）置 1。

在响应中断时，通过读 I2CRCV 的内容可以检测到中断的来源，从而确定地址是特定于器件的还是广播呼叫地址。

16.11 I^{2}C™ 主模式支持

用作主器件时，支持下列 6 个操作：

- 在 SDA 和 SCL 上产生启动条件。
- 在 SDA 和 SCL 上产生重复启动条件。
- 写 I2CTRN 寄存器，启动数据 / 地址的发送。
- 在 SDA 和 SCL 上产生停止条件。
- 配置 I^{2}C 端口，用以接收数据。
- 在数据字节接收完成后产生 $\overline{\text{ACK}}$ 条件。

16.12 I^{2}C™ 主模式的工作

主器件产生所有的串行时钟脉冲、启动和停止条件。出现停止条件或重复的启动条件，则传输结束。因为重复的启动条件是下一次串行传输的开始，此时不会释放 I^{2}C 总线。

在主发送模式下，串行数据通过 SDA 输出， SCL 输出串行时钟。发送的第一个字节是接收器件（从器件）地址（7 位）和数据方向位（R_W）。此时，数据方向位是逻辑 0。一次发送 8 位串行数据。在每个字节发送完之后，接收到一个 $\overline{\text{ACK}}$ 位。输出启动条件和停止条件，来表明串行传输的开始和结束。

在主接收模式下，发送的第一个字节是发送器件（从器件）地址（7 位）和数据方向位。此时，数据方向位（R_W）是逻辑 1。这样，主器件发送的第一个字节是一个 7 位从地址，后跟一个表明接收的 1。通过 SDA 接收串行数据，SCL 输出是串行时钟。一次接收 8 位串行数据。在每个字节接收之后，要发送一个 $\overline{\text{ACK}}$ 位。启动条件和停止条件，表明发送的开始和结束。

16.12.1 I²C™ 主发送

数据字节、 7 位地址或 10 地址后半部分的发送，通过简单地把值写入 I2CTRN 寄存器就能完成。当模块处于等待状态时，用户应该只写 I2CTRN。这将置 1 缓冲器满标志 （TBF），允许波特率发生器开始计数，并开始下一次发送。在 SCL 的下降沿后，地址 / 数据的每一个位将被移出至 SDA 引脚上。发送状态标志位 TRSTAT（I2CSTAT<14>）用来表明主器件是否正在进行发送。

16.12.2 I²C™ 主接收

通过设置接收使能位 RCEN（I2CCON<11>）可以使能主模式接收。在 RCEN 置 1 之前，I²C 模块必须是空闲的，否则 RCEN 位被忽略。波特率发生器开始计数，每次计满返回到零时，在每个时钟的上升沿，SCL 引脚状态发生翻转，数据移入至 I2CRSR 中。

16.12.3 波特率发生器 （BRG）

在 I²C 主模式下，波特率发生器的重载值位于 I2CBRG 寄存器中。波特率发生器装入该值后，发生器递减计数到 0，然后停止，直到发生再次装入。如果发生时钟伸裁，例如 SCL 引脚采样为高时，将重载波特率发生器。

根据 I²C 标准，FSCK 可以是 100 kHz 或 400 kHz 。然而，用户可以指定任何波特率，最高到 1 MHz。I2CBRG 的值不能是 0 或 1。

公式 16-1： I2CBRG 值

$$I2CBRG = \left(\frac{Fcy}{Fscl} - \frac{Fcy}{1,111,111}\right) - 1$$

16.12.4 时钟伸裁

在任何接收、发送或重复启动 / 停止条件期间，当主器件释放了 SCL 引脚 （允许 SCL 悬空为高电平），就会出现时钟伸裁。当释放 SCL 引脚时，波特率发生器暂停计数，直到 SCL 引脚被实际采样到高电平为止。当 SCL 引脚被采样到高电平时，波特率发生器重新装载 I2CBRG 的内容并开始计数。在外部器件使时钟保持为低时，这可以始终保证SCL高电平时间至少为一个BRG 计满返回计数周期。

16.12.5 多主器件通信、总线冲突与总线仲裁

多主器件工作的支持，是通过总线仲裁来实现的。当主器件在 SDA 引脚上输出地址 / 数据位时，第一个主器件使 SDA 悬空为高电平从而输出一个 “1”，而另一个主器件要输出 “0”，就会发生仲裁。当 SCL 引脚悬空为高时，数据应该是稳定的。如果 SDA 上预期数据是 “1”，但从 SDA 引脚采样到的数据是 “0”，那么就发生了总线冲突。主器件将置 1 MI2CIF 脉冲，并把 I²C 端口的主器件部分复位到空闲状态。

如果正在进行发送时产生总线冲突，将停止发送，清除 TBF标志，释放SDA和SCL线，而后可向I2CTRN写入数据。当用户执行 I²C 主事件中断服务程序时， 如果 I²C 总线空闲 （即 P 位置 1），用户可以通过产生一个启动条件继续通信。

当总线冲突发生在启动、重复启动、停止或应答条件期间，将中止这些条件，释放 SDA 和 SCL 线，清零 I2CCON 寄存器中对应的控制位。当用户执行总线冲突中断服务程序时，如果 I²C 总线空闲，用户可以通过产生一个启动条件继续通信。

主器件将继续监测 SDA 和 SCL 引脚，如果停止条件出现，将置 1 MI2CIF 位。

写 I2CTRN 将从第一个数据位开始数据的发送，无论总线发生冲突时发送停止在何处。

在多主器件环境中，在检测到启动和停止条件时产生中断，能够确定总线何时空闲。当 I2CSTAT 中 P 位置 1 时，器件可以控制 I²C 总线；否则，总线空闲，且 S 和 P 位清零。

16.13 CPU 休眠和空闲模式期间 I2C™ 模块的工作

16.13.1 CPU 休眠模式期间 I 2C™ 的工作

当器件进入休眠模式时，模块的全部时钟源都将关闭并保持在逻辑 0。如果休眠发生在发送过程中，且时钟停止时状态机部分进入发送，发送将中止。类似地，如果休眠出现在接收过程中，接收也将中止。

16.13.2 CPU 空闲模式期间 I2C™ 的工作

对于 I2C， I2CSIDL 位选择 CPU 空闲时模块是停止还是继续工作。如果 I2CSIDL = 0，模块将继续工作；如果 I2CSIDL = 1，模块将停止工作。

表 16-2：　I2C™ 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
I2CRCV	0200	—	—	—	—	—	—	—	—	接收寄存器								0000 0000 0000 0000
I2CTRN	0202	—	—	—	—	—	—	—	—	发送寄存器								0000 0000 1111 1111
I2CBRG	0204	—	—	—	—	—	—	—	波特率发生器									0000 0000 0000 0000
I2CCON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0001 0000 0000 0000
I2CSTAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000 0000 0000 0000
I2CADD	020A	—	—	—	—	—	—	地址寄存器										0000 0000 0000 0000

图注：　u = 未初始化位

注：　有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

17.0 通用异步收发器 （UART）模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）

本章介绍了通用异步接收器 / 发送器通信模块。

注：	由于 dsPIC30F2010 器件只拥有一个 UART，因此所有 Ux... 中的 x 仅指 x = 1。

17.1 UART 模块概述

UART 模块的主要特性是：

- 全双工 8 位或 9 位数据通信
- 偶校验、奇校验或无奇偶校验选项 （对于 8 位数据）
- 一或两个停止位
- 完全集成的波特率发生器，具有 16 位预分频器
- 在 30 MHz 指令速度时，波特率范围为 38 bps 到 1.875 Mbps
- 4 字深度发送数据缓冲器
- 4 字深度接收数据缓冲器
- 奇偶、帧和缓冲溢出错误检测
- 支持仅在地址检测 （第 9 位 = 1）时中断
- 独立的发送和接收中断
- 用于诊断支持的环回模式

图 17-1： UART 发送器框图

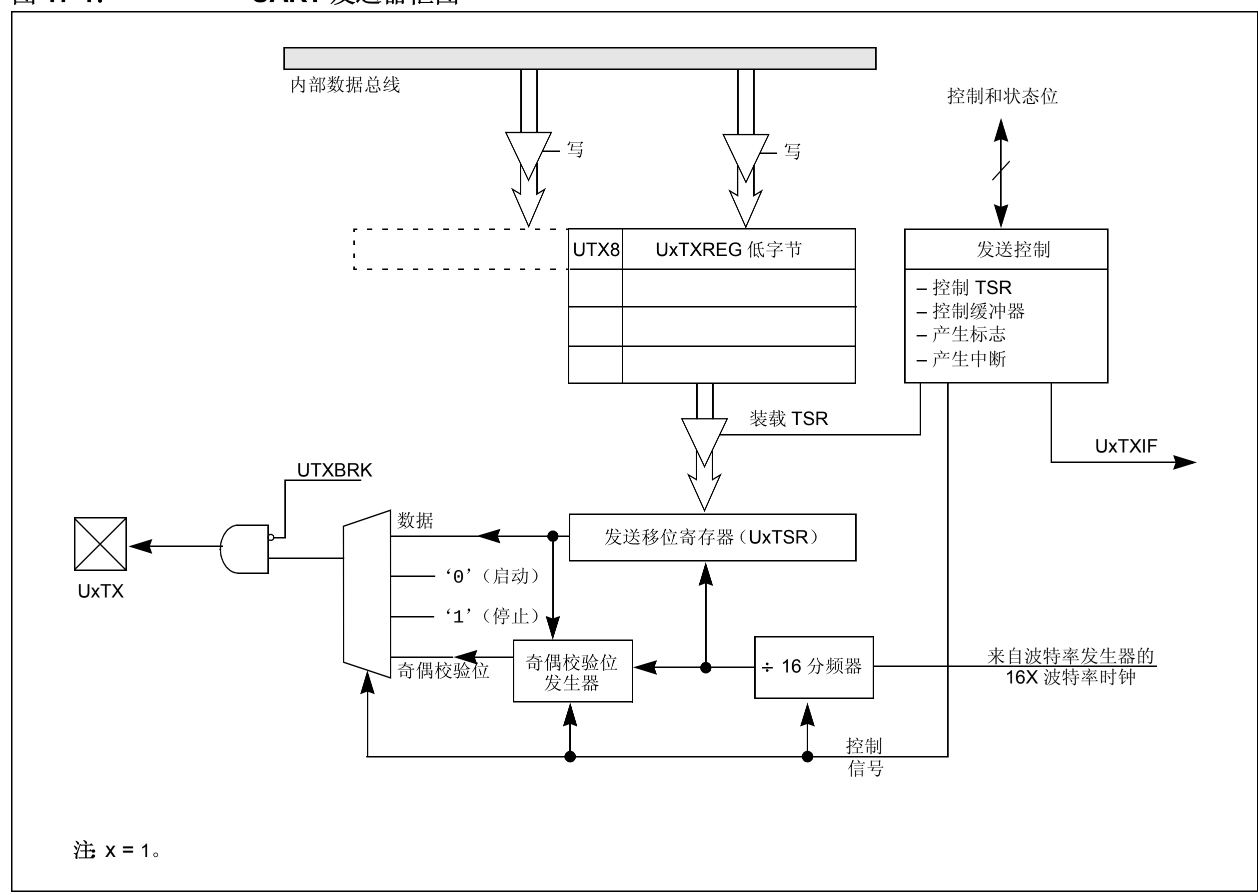
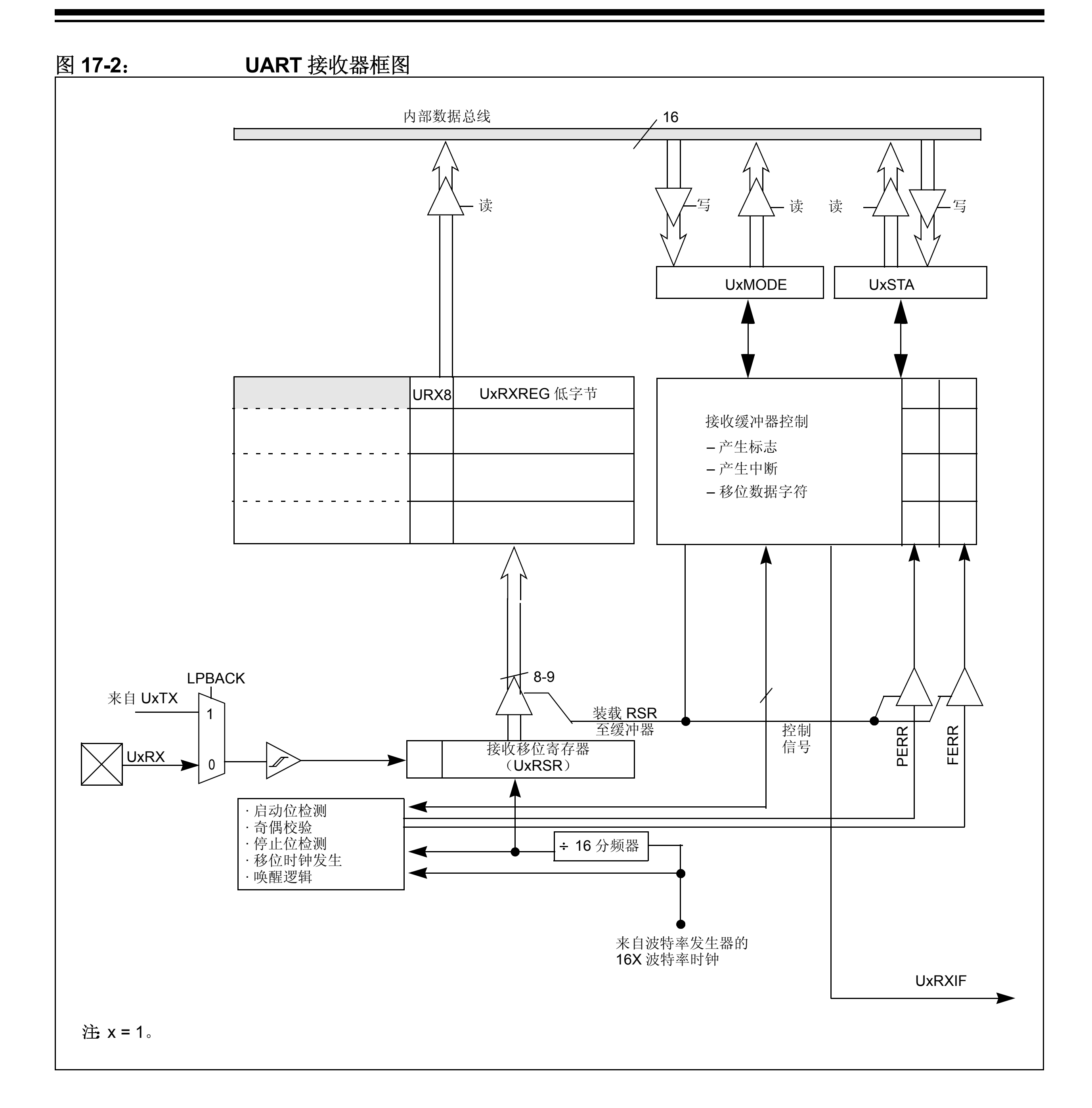


图 17-2： UART 接收器框图



17.2 使能和设置 UART

17.2.1 使能 UART

UART模块的使能，是通过置1 UxMODE寄存器（其中，x = 1）中的 UARTEN 位来进行的。一旦使能，UxTX 和 UxRX 引脚被分别配置为输出和输入，这改写了相应 I/O 端口引脚的 TRIS 和 LATCH 寄存器位的设置。在没有传输发生时， UxTX 引脚状态为逻辑 1。

17.2.2 禁止 UART

通过清零UxMODE寄存器中的UARTEN位，禁止UART模块。这是任何复位后的默认状态。如果 UART 被禁止，那么所有的 I/O 引脚均用作端口引脚，受相应端口引脚的锁存器和 TRIS 位的控制。

禁止 UART 模块将使缓冲器复位为空状态。缓冲器中的所有字符都将丢失，同时波特率计数器也将复位。

当 UART 模块禁止时，所有与之相关的错误和状态标志都将复位。URXDA、OERR、FERR、PERR、UTXEN、UTXBRK 和 UTXBF 位被清零，而 RIDLE 和 TRMT 被置 1。其他的控制位，包括 ADDEN、 URXISEL<1:0> 和 UTXISEL，以及 UxMODE 和 UxBRG 寄存器则不受影响。

当 UART 处于工作状态时，对 UARTEN 位清零将中止所有等待的发送和接收，同时还将像上面定义的那样将模块复位。再次使能 UART 将使用同样的配置重新启动 UART。

17.2.3 备用 I/O

通过置 1 ALTIO 位（UxMODE<10>）可使能备用 I/O 功能。如果 ALTIO = 1， UART 模块将使用 UxATX 和 UxARX 引脚（分别为备用发送引脚和备用接收引脚）而不是 UxTX 和 UxRX 引脚。如果 ALTIO = 0，UART 模块将使用 UxTX 和 UxRX 引脚。

17.2.4 设定数据、奇偶校验以及停止位选择

UxMODE 寄存器中的控制位 PDSEL<1:0>，用来选择发送时使用的数据长度和奇偶校验。数据长度可以是 8 位，具有偶校验位、奇校验位或没有校验位，数据长度也可以是没有校验位的 9 位。

STSEL 位决定在数据发送期间，是使用一个还是两个停止位。

UART 的默认（上电）设置是 8 位、无校验位以及 1 个停止位（通常表示为 8，N，1）。

17.3 发送数据

17.3.1 8 位数据模式下的发送

为了发送 8 位数据，必须执行下列步骤：

1. 设置 UART：
   首先，必须选择数据长度、奇偶校验和停止位的个数。然后，在 UxMODE 和 UxSTA 寄存器中设置发送和接收中断允许位以及优先级位。另外，必须把适当的波特率值写入 UxBRG 寄存器。
2. 通过置 1 UARTEN 位（UxMODE<15>），使能 UART。
3. 置 1 UTXEN 位（UxSTA<10>），从而使能发送。

注：	在 UARTEN 位被置 1 使能 UART 发送之后，必须置 1 UTXEN 位。

4. 向 UxTXREG 的低字节写入要发送的字节。它将被立即送往发送移位寄存器（UxTSR），在波特率时钟的下一个上升沿期间，串行位流将开始移出移位寄存器。另一种方法是，在 UTXEN = 0 时可写入数据字节，而后，用户可以把 UTXEN 置 1。这将使串行位流立即开始，因为波特率时钟将从清零状态启动。
5. 将产生发送中断，这取决于中断控制位 UTXISEL（UxSTA<15>）的值。

17.3.2 9 位数据模式下的发送

9 位数据发送涉及的步骤，类似于 8 位数据发送，但是必须将一个 16 位数据字（其中高 7 位始终为 0）写入 UxTXREG 寄存器。

17.3.3 发送缓冲器（UxTXB）

发送缓冲器为 9 位宽、4 字符深度。把发送移位寄存器（UxTSR）算上的话，用户实际上有一个 5 级深度的 FIFO（先进先出）缓冲器。 UTXBF 状态位（UxSTA<9>）表明发送缓冲器是否已满。

如果用户试图写已满的缓冲器，新数据将不会被 FIFO 接收，缓冲器内也不会有数据移位。这确保了能够从缓冲器溢出条件中恢复。

FIFO 在任何器件复位时复位，但当器件进入省电模式或从省电模式唤醒时， FIFO 不受影响。

17.3.4 发送中断

发送中断标志 （U1TXIF 或 U2TXIF）位于相应的中断标志寄存器中。

发送器产生一个边沿来置 1 UxTXIF 位。产生中断的条件取决于 UTXISEL 控制位：

a) 如果 UTXISEL = 0，当从发送缓冲器传输一个字到发送移位寄存器 （UxTSR）时，将产生中断。这意味着，发送缓冲器中至少有一个空字。
b) 如果 UTXISEL = 1，当从发送缓冲器传输一个字到发送移位寄存器 （UxTSR）且缓冲器为空时，将产生中断。

工作期间可以在这两种中断方式之间切换，有时这提供了更大的灵活性。

17.3.5 发送间隔符

把 UTXBRK 位 （UxSTA<11>）置 1，将使 UxTX 线被驱动为逻辑 0。UTXBRK 位改写了全部的发送器活动。从而，通常用户应该在置 1 UTXBRK 之前等待发送器空闲。

为了发送中止字符，必须由软件置 1 UTXBRK 位，而且保持该位置 1 至少 13 个波特率时钟周期。然后通过软件将 UTXBRK 位清零，产生停止位。在再次装载 UTXBUF 或开始其他发送活动前，用户必须等待至少一或两个波特率时钟周期，以便保证产生有效的停止位。

17.4 接收数据

17.4.1 在 8 位或 9 位数据模式下接收

在接收 8 位或 9 位数据时，必须执行下列步骤：

1. 设置 UART（见第 **17.3.1** 节 **"8 位数据模式下的发送"**）。
2. 使能 UART（见第 **17.3.1** 节 **"8 位数据模式下的发送"**）。
3. 当接收到一个或多个数据字时，将产生接收中断，这取决于接收中断的设置 （由 URXISEL 位（UxSTA<7:6>）指定）。
4. 读 OERR 位，判断是否产生溢出错误。OERR 位必须在软件中复位。
5. 从 UxRXREG 中读取接收数据。对 UxRXREG 的读操作，将使下一个字移动到接收 FIFO 的顶端，且将更新 PERR 和 FERR 值。

17.4.2 接收缓冲器 （UxRXB）

接收缓冲器是 4 字深缓冲器。加上接收移位寄存器（UxRSR），用户实际上有一个 5 字深的 FIFO 缓冲器。

URXDA （UxSTA<0>） = 1 表明接收缓冲器中还有数据。 URXDA = 0 意味着缓冲器为空。如果用户试图读空缓冲器，读取的是缓冲器中的原有值，在 FIFO 中不会产生数据移位。

任何器件复位时， FIFO 都将复位。当器件进入省电模式或从省电模式中唤醒时，它不受影响。

17.4.3 接收中断

可以从相应的中断标志寄存器中读取接收中断标志（U1RXIF） 。中断标志由接收器产生的边沿来进行置 1。置 1 接收器中断标志的条件，取决于 URXISEL<1:0>（UxSTA<7:6>）控制位的设定。

a) 如果URXISEL<1:0> = 00或01，每当一个数据字从接收移位寄存器 （UxRSR）送往接收缓冲器后就会产生中断。接收缓冲器中可以有一个或多个字符。
b) 如果 URXISEL<1:0> = 10，当一个字从接收移位寄存器 （UxRSR）送往接收缓冲器，使得缓冲器中有 3 个字符时，就会产生中断。
c) 如果 URXISEL<1:0> = 11，当一个字从接收移位寄存器 （UxRSR）送往接收缓冲器，使得缓冲器中有 4 个字符 （即缓冲器满）时，就会产生中断。

工作期间可以在这两种中断方式之间切换，尽管正常工作期间这样做并不可取。

17.5 接收错误处理

17.5.1 接收缓冲器溢出错误 （OERR 位）

如果满足下列全部条件，OERR 位（UxSTA<1>）将置 1：

a) 接收缓冲器满。
b) 接收移位寄存器满，但不能把字符送往接收缓冲器。
c) 检测到 UxRSR 中字符的停止位，这表明 UxRSR 需要把字符送往缓冲器。

一旦 OERR 置 1，就不会有数据移入 UxRSR （除非在软件中将 OERR 位清零，或者发生了复位） 。存放在 UxRSR 和 UxRXREG 中的数据继续有效。

17.5.2 帧错误 （FERR）

如果检测到0而不是停止位，则FERR位（UxSTA<2>）置1。如果选用两个停止位，则两个停止位都必须是1，否则将置1 FERR。FERR位是只读的，它将和收到的数据一起被送入缓冲器。任何复位时都将清零 FERR位。

17.5.3 奇偶校验错误 （PERR）

如果接收到的字奇偶校验错误的话，PERR 位（UxSTA<3>）将置1。此错误位仅在选取了奇偶校验模式（奇或偶）时适用。PERR 位是只读的，它将和收到的数据一起被送入缓冲器。任何复位时都将清零PERR位。

17.5.4 空闲状态

当接收器处于活动状态（即，在初次检测到启动位与停止位完成之间时），RIDLE位（UxSTA<4>）为0。在停止位完成与检测到下一个启动位之间时，RIDLE位为1，表明UART空闲。

17.5.5 接收中止符

接收器会根据在PDSEL（UxMODE<2:1>）和STSEL（UxMODE<0>）中设置的值，计数并等待一定的位时间数。

如果中止超过13个位时间，则在PDSEL和STSEL指定的位时间数之后，认为接收完成。URXDA 位置 1，FERR置1，接收FIFO中装载零，同时产生中断（若允许中断）且RIDLE位置1。

如果模块收到长中止信号，同时接收器检测到了启动位、数据位和无效的停止位（这将把FERR置1），那么接收器在找寻下一个启动位之前，必须等待有效停止位出现。接收器不能假定线上的中止条件就是下一个启动位。

中止符被视为一个全0的字符，且FERR位置1。中止符将被装载到缓冲器中。除非接收到停止位，不会再进行接收。注意，当停止位还没有接收到时，RIDLE将置1。

17.6 地址检测模式

置1 ADDEN位（UxSTA<5>）来使能这个特殊的模式，在此模式下，如果第9位（URX8）值为1，则表明接收到的字是一个地址，而不是数据。此模式仅适用于 9 位数据通信。在此模式下，URXISEL 控制位对中断产生没有任何影响，因为中断（如果允许的话）仅在接收到的字的第9位为1时产生。

17.7 环回模式

置 1 LPBACK 位来使能这个特殊的模式，在该模式下UxTX 引脚从内部连接到 UxRX 引脚。当配置为环回模式时，UxRX 引脚与内部 UART 接收逻辑断开。但是，UxTX 引脚仍然正常工作。

要选择该模式：

a) 将UART配置为所需的工作模式。
b) 设置LPBACK = 1，使能环回模式。
c) 使能发送，如同**第17.3节“发送数据”**中定义的那样。

17.8 波特率发生器

UART 有一个 16 位波特率发生器，以便在波特率发生时获得最大的灵活性。波特率发生器寄存器（UxBRG）是可读写的。波特率的计算如下：

BRG = UxBRG寄存器中的16位值 （0到65535）

F_{CY} = 指令时钟速度 （$1/T_{CY}$）

波特率由公式 17-1 给出。

公式 17-1： 波特率

$$波特率 = F_{CY} / (16*(BRG + 1))$$

从而，可能的最大波特率是

$F_{CY}/16$ （如果BRG = 0），

可能的最小波特率是

$F_{CY} / (16* 65536)$。

对于在30 MIPS工作的16位波特率发生器，能够达到的最小波特率是28.5 bps。

17.9 自动波特率支持

为了让系统确定接收字符的波特率，可选择将输入连接到所选择的捕捉输入通道。为使能该模式，用户必须对输入捕捉模块进行设置，以检测启动位的上升沿和下降沿。

17.10 UART 在 CPU 休眠和空闲模式下的工作

17.10.1 UART 在 CPU 休眠模式下的工作

当器件进入休眠模式时，模块的所有时钟源都将关闭，保持在逻辑 0 状态。如果在发送进行期间进入休眠模式的话，发送将中止，并且 UxTX 引脚被驱动为逻辑 1。类似地，如果接收进行期间进入休眠模式，接收将中止。休眠模式不影响 UxSTA、 UxMODE、发送和接收的寄存器和缓冲器以及 UxBRG 寄存器。

器件进入休眠模式前，如果 WAKE 位（UxMODE<7>）置 1，那么 UxRX 引脚上的下降沿将产生接收中断。接收中断选择模式位（URXISEL）对该功能没有影响。如果接收中断允许，那么这将把器件从休眠中唤醒。为了产生唤醒中断， UARTEN 位必须置 1。

17.10.2 UART 在 CPU 空闲模式下的工作

对于 UART，USIDL 位选择当器件进入空闲模式时，模块是停止还是继续工作。如果 USIDL = 0，模块将在空闲模式期间继续工作。如果 USIDL = 1，模块将在空闲模式下停止工作。

表 **17-1**： **UART1** 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
U1MODE	020C	UARTEN	—	USIDL	—	—	ALTIO	—	—	WAKE	LPBACK	ABAUD	—	—	PDSEL1	PDSEL0	STSEL	0000 0000 0000 0000
U1STA	020E	UTXISEL	—	—	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0000 0001 0001 0000
U1TXREG	0210	—	—	—	—	—	—	—	UTX8	发送寄存器								0000 000u uuuu uuuu
U1RXREG	0212	—	—	—	—	—	—	—	URX8	接收寄存器								0000 0000 0000 0000
U1BRG	0214	波特率发生器预分频器																0000 0000 0000 0000

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

18.0 10 位高速模数转换器 （ADC）模块

> **注：**本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。

10 位高速模数转换器 （ADC）可将模拟输入信号转换为 10 位数字值。该模块基于逐次逼近寄存器（Successive Approximation Register，SAR）结构，可实现最大为 500 ksps 的采样速率。ADC 模块拥有多达 16 个模拟输入通道，这些模拟输入通过多路开关连接到 4 个采样 / 保持放大器。采样 / 保持电路的输出输入到转换器，经转换器转换后得到数字结果。可通过软件选择模拟参考电压为器件供电电压（AVDD/AVSS）或（VREF+/VREF-）引脚上的电压。ADC 具有一个独特的功能，当器件处于休眠模式时仍能工作。

ADC 模块具有 6 个 16 位寄存器：

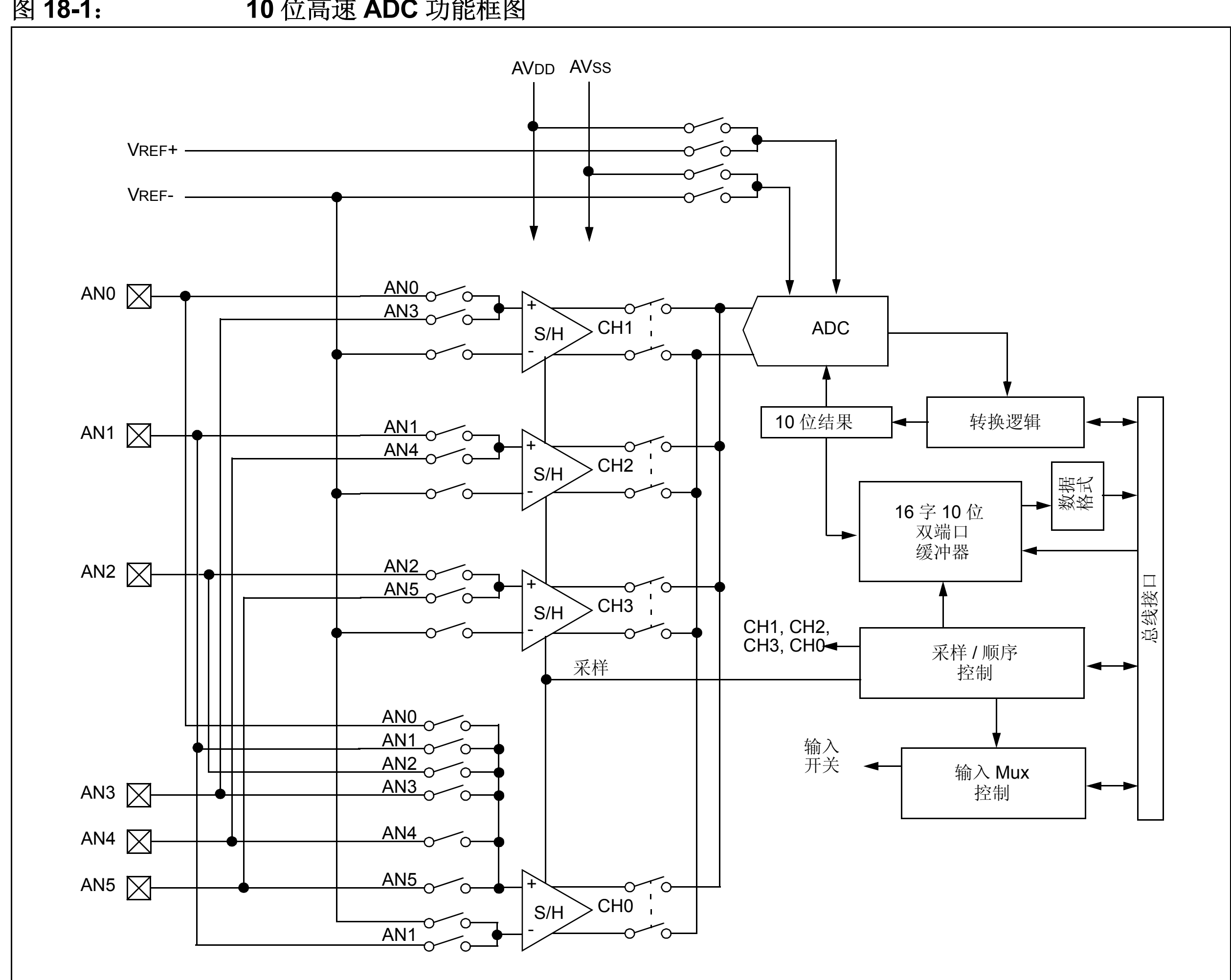
- A/D 控制寄存器 1 （ADCON1）
- A/D 控制寄存器 2 （ADCON2）
- A/D 控制寄存器 3 （ADCON3）
- A/D 输入选择寄存器 （ADCHS）
- A/D 端口配置寄存器 （ADPCFG）
- A/D 输入扫描选择寄存器 （ADCSSL）

ADCON1、ADCON2 和 ADCON3 寄存器控制 ADC 的工作。ADCHS 寄存器选择要转换的输入通道。ADPCFG 寄存器用于将端口引脚配置为模拟输入或数字 I/O。ADCSSL 寄存器用于选择要扫描的输入通道。

> **注：** 当 ADON = 1 时，不应对 SSRC<2:0>、ASAM、SIMSAM、SMPI<3:0>、BUFM 和 ALTS 位以及 ADCON3 和 ADCSSL 寄存器进行写入操作，否则将导致不确定的结果。

图 18-1 给出了 ADC 的框图。

图 18-1： 10 位高速 ADC 功能框图



18.1 A/D 结果缓冲器

模块包含一个称为 ADCBUF0...ADCBUFF 的 16 字双端口只读缓冲器，对 A/D 转换结果进行缓冲。 RAM 为 10 位宽，但读出为不同格式的 16 位字。用户软件不能对 16 个 A/D 转换结果缓冲寄存器 ADCBUF0 至 ADCBUFF 的内容进行写入操作。

18.2 转换操作

当 ADC 模块配置完毕后，置 1 SAMP 位将启动采样操作。可编程位、定时器超时和外部事件等触发源将终止采集并启动转换。当 A/D 转换完成时，结果将被装入 ADCBUF0...ADCBUFF，且在 SMPI 位指定的采样数之后 A/D 中断标志位 ADIF 和 DONE 位将被置 1。

进行 A/D 转换时，应遵守以下步骤：

1. 配置 ADC 模块：
   - 配置模拟引脚、参考电压和数字 I/O
   - 选择 A/D 输入通道
   - 选择 A/D 转换时钟
   - 选择 A/D 转换触发源
   - 开启 ADC 模块
2. 配置 A/D 中断 （如果需要）：
   - 清零 ADIF 位
   - 选择 A/D 中断优先级
3. 启动采样。
4. 等待所需的采集时间。
5. 触发采样结束，启动转换
6. 等待 A/D 转换完成，可通过以下方式实现：
   - 等待 A/D 中断
   - 等待 DONE 位被置 1
7. 读 A/D 结果缓冲器，清零 ADIF （如果需要）。

18.3 选择转换顺序

几组控制位可用来对 A/D 将输入连接至采样 / 保持通道、转换通道、写入缓冲器存储器和产生中断的过程进行选择控制。该过程将由采样时钟进行控制。

SIMSAM 位对多个通道的采样 / 转换顺序进行控制。如果 SIMSAM 位为 ‘0’，将在两个或四个采样时钟内对两个或四个选择的通道按顺序进行采样和转换。如果 SIMSAM 位为 ‘1’，将在一个采样时钟内同时对两个或四个选择的通道进行采样。然后按顺序转换通道。显然，如果只选择 1 个通道， SIMSAM 位将不适用。

CHPS 位用于选择对多少个通道进行采样。通道数目可为 1、 2 或 4 个通道不等。如果 CHPS 选择 1 个通道，则将在一个采样时钟内对 CH0 通道进行采样并转换。结果将存放在缓冲器中。如果 CHPS 选择 2 个通道，将对 CH0 和 CH1 通道进行采样和转换。如果 CHPS 选择 4 个通道，则将对 CH0、CH1、CH2 和 CH3 通道进行采样和转换。

SMPI 位用于选择在中断发生前进行采集 / 转换的次数。该数值可为每次中断进行采样 1 至 16 次。

用户不能将 CHPS 和 SMPI 位的组合编程设定为指定每次中断进行 16 次或 8 次转换，具体是 16 次还是 8 次取决于 BUFM 位的状态。当置 1 时，BUFM 位将 16 字结果缓冲器 （ADCBUF0...ADCBUFF）分为两个 8 字缓冲器。每个中断事件将交替写入 8 字缓冲器。BUFM 位的使用取决于中断之后有多少时间将数据从缓冲器内移出，而这取决于具体的应用。

如果处理器可在对一个通道进行采样和转换的时间内快速卸空一个满的缓冲器，则 BUFM 位可为 ‘0’ 且每次中断可进行最多 16 次转换。处理器在一次采样和转换的时间内移出 16 个转换结果。

如果在采样和转换时间内处理器不能卸空缓冲器，BUFM 位应为 ‘1’ 。例如，如果 SMPI<3:0>（ADCON2<5:2>）= 0111，则先将 8 个转换结果装入其中一个 8 字缓冲器，此后将产生中断。再将另外的 8 个转换结果装入另一个 8 字缓冲器。处理器将利用相邻中断之间的全部时间来移出 8 个转换结果。

ALTS 位用来在采样过程中轮换选择的输入。输入多路开关具有两组采样输入：MUX A 和 MUX B。如果 ALTS 位为 ‘0’，只选择 MUX A 输入进行采样。如果 ALTS 位为 ‘1’ 且 SMPI<3:0> = 0000，在第一个采样 / 转换过程将选择 MUX A 输入通道，而在下一个采样 / 转换过程，将选择 MUX B 输入通道。

CSCNA 位（ADCON2<10>）允许对 CH0 通道输入 MUX A 组的选定数量的模拟输入进行顺序扫描。ADCSSL 寄存器选择要扫描的输入。如果 ADCSSL 寄存器中的某个位为 ‘1’，则将选择相应的输入。将在每次中断之后，始终按照输入编号从低至高的顺序对输入进行扫描。如果选择的输入数目大于每次中断进行的采样数，则编号高于采样数的输入将不会被使用。

18.4 转换触发的编程设定

转换触发将终止数据采集和并开始进行要求的转换。

SSRC<2:0> 位用于选择转换触发源。

SSRC 位提供多达 5 个备用的转换触发源。

当 SSRC<2:0> = 000 时，转换触发由软件进行控制。清零 SAMP 位将导致触发转换。

当 SSRC<2:0> = 111 （自动启动模式）时，转换触发由 A/D 时钟进行控制。 SAMC 位用于选择采样起始和转换起始之间的 A/D 时钟数。这将提供最快的多通道转换速率。 SAMC 必须始终大于等于 1 个时钟周期。

其他触发源可能来自定时器模块、电机控制 PWM 模块或外部中断。

注：	如果需要使A/D工作在最大指定转换速率，应选择自动转换触发选项 （SSRC = 111）且自动采样时间应该设定为 1 个 TAD （SAMC = 00001）。此配置将使全部转换周期 （采样 + 转换）为 13 个 TAD。 使用任何其他转换触发方式将导致额外的 TAD 周期以与 A/D 的外部事件同步。

18.5 中止转换

转换过程中清零ADON位将中止当前的转换并停止采样过程。不会用部分完成的 A/D 采样转换来更新 A/D 结果寄存器对。即，对应的 ADCBUF 缓冲器单元将仍然保持上一次转换完成后的值 （即上一次写入该缓冲器的值）。

如果清零 ADON 位与自动启动同时发生，清零 ADON 位具有更高的优先级。

A/D 转换中止后，在置 1 SAMP 位启动下一次采样之前需等待 2 个 TAD 的时间。

如果指定进行顺序采样，A/D 将在与下一个要转换通道对应的下一个采样脉冲处继续工作。如果指定同时采样， A/D 将继续下一个多通道组转换过程。

18.6 选择 A/D 转换时钟

A/D 转换需要 12 个 TAD。通过软件方式使用 6 位计数器可选择 A/D 转换的时钟源。TAD 存在 64 种可能的选择。

公式 18-1： A/D 转换时钟

$$T_{AD} = T_{CY} * (0.5*(ADCS<5:0> + 1))$$

$$ADCS<5:0> = 2 \frac{T_{AD}}{T_{CY}} - 1$$

通过置 1 ADRC 位可选择内部 RC 振荡器。

为使 A/D 转换正确，必须选择合适的 A/D 转换时钟（TAD）以确保最小 83.33 ns （当 VDD = 5V 时）的 TAD 时间。其他工作条件下的最小 TAD，可参见第 **22.0 节 “电气特性”**。

例 18-1 中显示了 ADCS<5:0> 位的计算示例，其中假定器件工作速度为 30 MIPS。

例 18-1： A/D 转换时钟计算

T_{AD} = 84 ns
T_{CY} = 33 ns（30 MIPS）

$$ADCS<5:0> = 2 \frac{T_{AD}}{T_{CY}} - 1 = 2 \bullet \frac{84\ ns}{33\ ns} - 1 = 4.09$$

因此，
设置 ADCS<5:0> = 5

$$实际\ T_{AD} = \frac{T_{CY}}{2}(ADCS<5:0> + 1) = \frac{33\ ns}{2}(5 + 1) = 99\ ns$$

18.7 A/D 转换速度

dsPIC30F 的 10 位 ADC 规范允许最大 1 Msps 的采样速率。表 18-1 汇总了 dsPIC30F 10 位 ADC 的转换速度及其要求的工作条件。

表 18-1： 10 位 A/D 转换速率参数

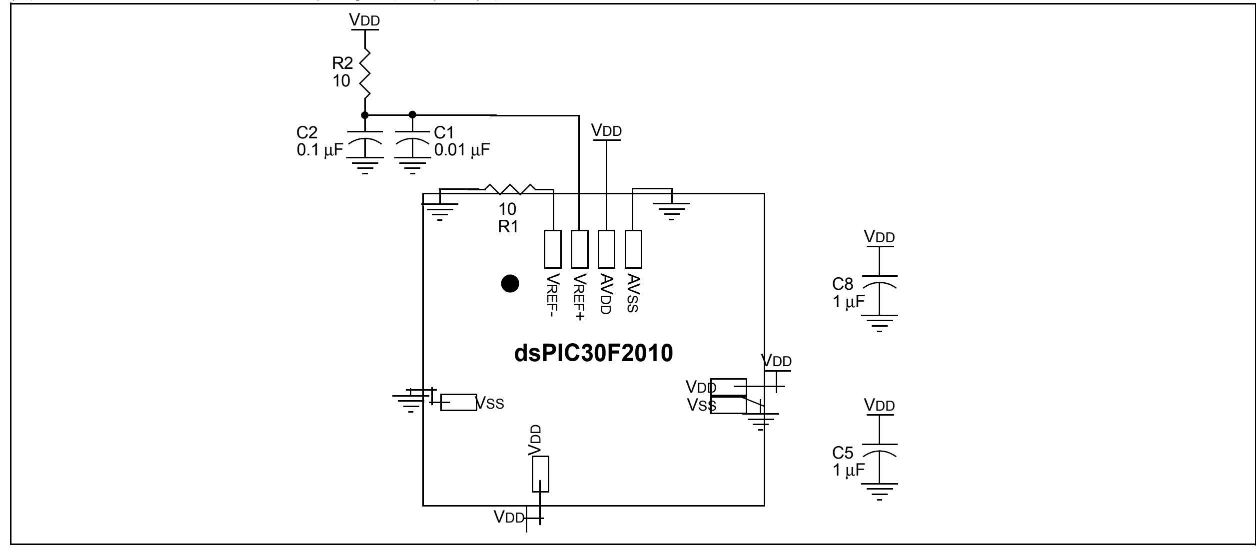
dsPIC30F 10 位 A/D 转换器转换速率						
A/D 速度	**TAD** 最小值	采样时间 最小值	**RS** 最大值	**VDD**	温度	**A/D 通道配置**
最大 1 Msps(1)	83.33 ns	12 TAD	500Ω	4.5V 至 5.5V	-40°C 至 +85°C	VREF- VREF+ ANx CH1、CH2 或 CH3 S/H CH0 S/H ADC
最大 750 ksps(1)	95.24 ns	2 TAD	500Ω	4.5V 至 5.5V	-40°C 至 +85°C	VREF- VREF+ ANx CHx S/H ADC
最大 600 ksps(1)	138.89 ns	12 TAD	500Ω	3.0V 至 5.5V	-40°C 至 +125°C	VREF- VREF+ ANx CH1、CH2 或 CH3 S/H CH0 S/H ADC
最大 500 ksps	153.85 ns	1 TAD	5.0 kΩ	4.5V 至 5.5V	-40°C 至 +125°C	VREF- 或 AVSS VREF+ 或 AVDD ANx CHx S/H ADC ANx 或 VREF-
最大 300 ksps	256.41 ns	1 TAD	5.0 kΩ	3.0V 至 5.5V	-40°C 至 +125°C	VREF- 或 AVSS VREF+ 或 AVDD ANx CHx S/H ADC ANx 或 VREF-

注 1：为确保正常工作，必须使用外部 VREF- 和 VREF+ 引脚。推荐电路请参见图 18-2。

配置指导给出了转换速度高于 500 ksps 时要求的设置值，因为这些转换速度需要使用外部 VREF 引脚，且配置过程有一些差别。此处忽略了与转换速度关系不大的配置细节。

图 18-2 给出了转换速度高于 500 ksps 时的推荐电路。

图 18-2： ADC 参考电压原理图



18.7.1 1 Msps 时的配置指南

1 Msps 时的配置取决于要采样单个输入引脚还是多个引脚。

18.7.1.1 单路模拟输入

对于 1 Msps 单路模拟输入的转换，必须至少使能两个采样 / 保持 （S/H）通道。必须将模拟输入多路开关配置为将同一个输入引脚连接到两个采样 / 保持通道。在 A/D 对一个 S/H 通道保持的值进行转换的同时，另外一个 S/H 通道采集新的输入采样。

18.7.1.2 多路模拟输入

A/D 转换器可用于使用多个采样 / 保持通道对多路模拟输入进行采样。在这种情况下，不同的输入信号共享总的 1 Msps 的采样速率。例如，可以对 4 路输入进行采样，每个信号的采样速率为 250 ksps ；或对 2 路输入进行采样，每个信号的采样速率为 500 ksps。在这种配置下，必须采用顺序采样以使每路输入具有足够的采样时间。

18.7.1.3 1 Msps 时的配置

为获得 1 Msps 的转换速率，需要进行如下配置。

- 符合表 19-2 中提供的条件
- 按照图 18-2 所示的推荐电路连接外部 VREF+ 和 VREF- 引脚
- 设置 ADCON1 寄存器的 SSRC<2:0> = 111 使能自动转换选项
- 将ADCON1寄存器的ASAM控制位置1使能自动采样
- 将 ADCON1 寄存器的 SIMSAM 位清零使能顺序采样
- 通过写ADCON2寄存器的CHPS<1:0>控制位来使能至少两个采样 / 保持通道
- 写 ADCON2 寄存器的 SMPI<3:0> 控制位来设置两次中断之间所需的转换次数。至少要设置 SMPI<3:0> = 0001，因为至少要使能两个采样 / 保持通道
- 通过写ADCON3寄存器的ADCS<5:0>控制位来将 A/D 时钟周期配置为：

$$\frac{1}{12 \times 1{,}000{,}000} = 83.33 \text{ ns}$$

- 通过写 SAMC<4:0> = 00010 来将采样时间配置为 2 TAD
- 通过写 ADCHS 寄存器为每个模拟输入引脚选择最少两个通道

18.7.2 750 ksps 时的配置指南

为获得 750 ksps 的转换速率，需要进行如下配置。配置假设对单路模拟输入进行采样。

- 符合表 19-2 中提供的条件
- 按照图 18-2 所示的推荐电路连接外部 VREF+ 和 VREF- 引脚
- 设置 ADCON1 寄存器的 SSRC<2:0> = 111 使能自动转换选项
- 将ADCON1寄存器的ASAM控制位置1使能自动采样
- 通过设置ADCON2寄存器的CHPS<1:0> = 00来使能一个采样 / 保持通道
- 写 ADCON2 寄存器的 SMPI<3:0> 控制位来设置两次中断之间所需的转换次数。
- 通过写ADCON3寄存器的ADCS<5:0>控制位来将 A/D 时钟周期配置为：

$$\frac{1}{(12 + 2) \text{ X } 750{,}000} = 95.24 \text{ ns}$$

- 通过写 SAMC<4:0> = 00010 来将采样时间配置为 2 TAD

18.7.3 600 ksps 时的配置指南

600 ksps时的配置取决于要采样单个输入引脚还是多个引脚。

18.7.3.1 单路模拟输入

以600 ksps对单路模拟输入进行转换时，必须至少使能两个采样 / 保持 （S/H）通道。必须将模拟输入多路开关配置为将同一个输入引脚连接到两个采样 / 保持通道。在 A/D 对一个 S/H 通道保持的值进行转换的同时，另外一个 S/H 通道采集新的输入采样。

18.7.3.2 多路模拟输入

A/D 转换器可用于使用多个采样 / 保持通道对多路模拟输入进行采样。在这种情况下，不同的输入信号共享总的 600 ksps 的采样速率。例如，可以对 4 路输入进行采样，每个信号的采样速率为 150 ksps ；或对 2 路输入进行采样，每个信号的采样速率为 300 ksps。在这种配置下，必须采用顺序采样以使每路输入具有足够的采样时间。

18.7.3.3 600 ksps 时的配置

为获得 600 ksps 的转换速率，需要进行如下配置。

- 符合表 19-2 中提供的条件
- 按照图 18-2 所示的推荐电路连接外部 VREF+ 和 VREF- 引脚
- 设置 ADCON1 寄存器的 SSRC<2:0> = 111 使能自动转换选项
- 将ADCON1寄存器的ASAM控制位置1使能自动采样
- 将 ADCON1 寄存器的 SIMSAM 位清零使能顺序采样
- 通过写ADCON2寄存器的CHPS<1:0>控制位来使能至少两个采样 / 保持通道
- 写 ADCON2 寄存器的 SMPI<3:0> 控制位来设置两次中断之间所需的转换次数。至少要设置 SMPI<3:0> = 0001，因为至少要使能两个采样 / 保持通道
- 通过写ADCON3寄存器的ADCS<5:0>控制位来将 A/D 时钟周期配置为：

$$\frac{1}{12 \text{ x } 600{,}000} = 138.89 \text{ ns}$$

- 通过写 SAMC<4:0> = 00010 来将采样时间配置为 2 TAD
- 通过写 ADCHS 寄存器为每个模拟输入引脚选择最少两个通道

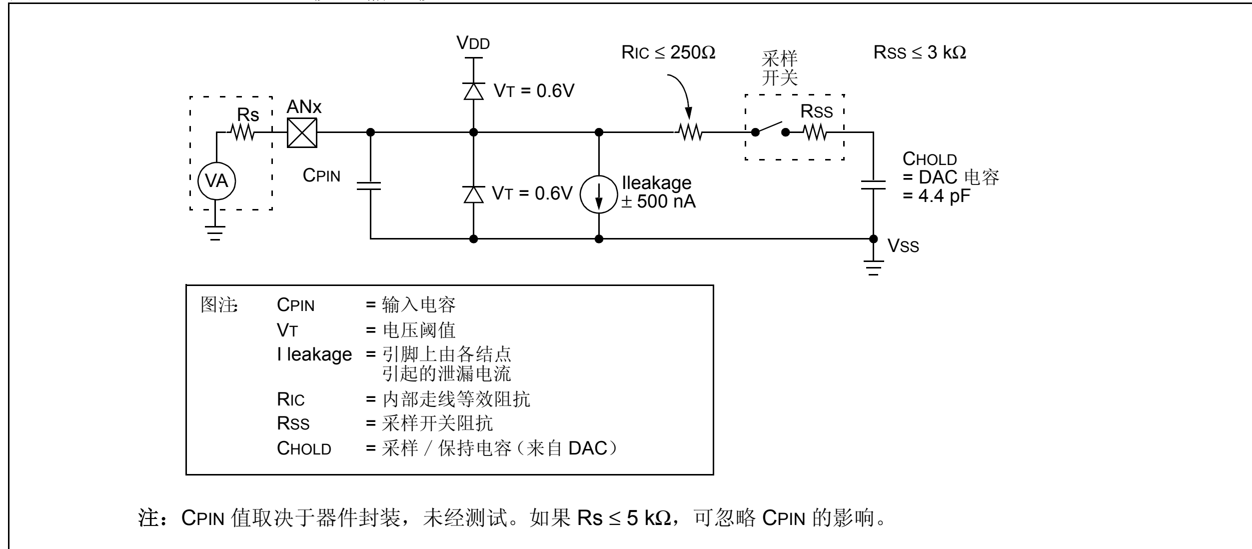
18.8 A/D 采集要求

图 18-3 给出了 10 位 A/D 转换器的模拟输入模型。A/D 的总采样时间是内部放大器稳定时间、器件 V_{DD} 和保持电容充电时间的函数。

为了使 A/D 转换器达到规定的精度，必须让充电保持电容（C_{HOLD}）充分充电至模拟输入引脚上的电平。信号源阻抗（R_S）、内部走线等效阻抗（R_{IC}）和内部采样开关阻抗（R_{SS}）共同地直接影响电容 C_{HOLD} 充电所需的时间，所以模拟信号源的总阻抗应足够小，以便在选择的采样时间内对保持电容充分充电。为了把引脚泄漏电流对 A/D 转换器精度的影响降到最低，最大建议信号源阻抗 R_S 为 5 kΩ。选择（改变）了模拟输入通道后，采样工作必须在启动转换前完成。在每次采样操作前，内部保持电容将处于放电状态。

两次转换之间应该至少留出 1 个 T_{AD} 周期作为采样时间 T_{SAMP}。可通过软件手动方式设定/清零SAMP位以对该采样时间进行控制，也可由 A/D 转换器进行自动控制。对于自动配置方式，用户必须在转换触发之间保留足够时间以满足最小采样时间。有关 T_{AD} 和采样时间的要求，参见器件电气规范。

图 18-3： ADC 模拟输入模型



注：C_{PIN} 值取决于器件封装，未经测试。如果 $R_S \leq 5$ kΩ，可忽略 C_{PIN} 的影响。

18.9 模块掉电模式

模块具有 3 种内部功耗模式。当 ADON 位为 ‘1’ 时，模块将处于工作模式，模块处于完全供电和工作状态；当 ADON 为 ‘0’ 时，模块将处于关断模式。电路的数字和模拟部分将被禁止以最大程度地降低电流消耗。为从关断模式返回工作模式，用户必须等待 ADC 电路进入稳定状态。

18.10 CPU 休眠和空闲模式下的 A/D 操作

18.10.1 CPU 休眠模式下的 A/D 操作

当器件进入休眠模式时，模块的所有时钟源将被关闭且停留在逻辑 ‘0’ 状态。

如果在转换过程中进入休眠模式，转换将中止。当器件退出休眠模式时，转换器将不会继续进行部分完成的转换。

器件进入或退出休眠模式将不会对寄存器内容造成影响。

如果 A/D 时钟源设定为 RC （ADRC = 1），则 ADC 模块可在休眠模式下进行工作。当选择RC时钟源时，ADC 在启动转换之前将等待一个指令周期。其间可执行 SLEEP 指令，这将消除转换过程中所有的数字开关噪声。当转换操作完成之后，DONE 位将置 1 且转换结果将被装入 ADCBUF 寄存器。

如果允许 A/D 中断，则器件将从休眠状态唤醒。如果 A/D 中断被禁止，ADC 模块将被关闭，尽管 ADON 位仍然保持置 1 状态。

18.10.2 CPU 空闲模式下的 A/D 操作

ADSIDL位选择ADC模块在空闲模式下是停止工作还是继续工作。如果 ADSIDL = 0，当器件进入空闲模式时，该模块将继续工作。如果 ADSIDL= 1，在空闲模式下该模块将停止工作。

18.11 复位的影响

器件复位强制所有寄存器进入复位状态。这将迫使ADC模块关闭并中止任何正在进行的转换。 ADCBUF 寄存器中的值将不会被修改。上电复位时， A/D 结果寄存器将包含未知数据。

18.12 输出格式

A/D结果为10位宽，数据缓冲器RAM也是10位宽。当对该缓冲器执行读操作时，读出数据将以四种不同格式之一进行表示。 FORM<1:0> 位用于选择格式。每一种输出格式将读出数据转换为数据总线上的 16 位结果。

写数据总是采用右对齐 （整数）的格式。

图 18-4： A/D 输出数据格式

RAM 内容：							d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
读至总线：																
有符号小数 （1.15）	$\overline{\text{d09}}$	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
小数 （1.15）	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
有符号整数	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	d08	d07	d06	d05	d04	d03	d02	d01	d00
整数	0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00

18.13 配置模拟端口引脚

ADPCFG 和 TRIS 寄存器用于控制 A/D 端口引脚的操作。当端口引脚作为模拟输入时，必须将其对应的 TRIS 位置 1 （输入）。如果 TRIS 位被清零 （输出），将对数字输出电平 （V_{OH} 或 V_{OL}）进行转换。

A/D 操作将不受 CH0SA<3:0>/CH0SB<3:0> 位以及 TRIS 位状态的影响。

当对 PORT 寄存器进行读操作时，所有配置为模拟输入通道的引脚将被读为零。

配置为数字输入的引脚将不会对模拟输入进行转换。任何定义为数字输入的引脚 （包括 ANx 引脚）上的模拟电平可能导致输入缓冲器消耗电流超过器件规范限定值。

18.14 连接注意事项

模拟输入与 V_{DD} 和 V_{SS} 之间连接有二极管作为 ESD 保护。这就要求模拟输入电压必须介于 V_{DD} 和 V_{SS} 之间。如果输入电压超出此范围 0.3V 以上（任一方向上），就会有一个二极管正偏，而且如果超过输入电流规范可能会损坏器件。

有时可外接 RC 滤波器来对输入信号进行抗混叠滤波。R 元件的选择要确保满足采样时间要求。任何通过高阻抗连接到模拟输入引脚上的外部元件（如电容和齐纳二极管等）在引脚上的泄漏电流都应极小。

表 18-2：　ADC 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
ADCBUF0	0280	—	—	—	—	—	—	ADC 数据缓冲器 0										0000 00uu uuuu uuuu
ADCBUF1	0282	—	—	—	—	—	—	ADC 数据缓冲器 1										0000 00uu uuuu uuuu
ADCBUF2	0284	—	—	—	—	—	—	ADC 数据缓冲器 2										0000 00uu uuuu uuuu
ADCBUF3	0286	—	—	—	—	—	—	ADC 数据缓冲器 3										0000 00uu uuuu uuuu
ADCBUF4	0288	—	—	—	—	—	—	ADC 数据缓冲器 4										0000 00uu uuuu uuuu
ADCBUF5	028A	—	—	—	—	—	—	ADC 数据缓冲器 5										0000 00uu uuuu uuuu
ADCBUF6	028C	—	—	—	—	—	—	ADC 数据缓冲器 6										0000 00uu uuuu uuuu
ADCBUF7	028E	—	—	—	—	—	—	ADC 数据缓冲器 7										0000 00uu uuuu uuuu
ADCBUF8	0290	—	—	—	—	—	—	ADC 数据缓冲器 8										0000 00uu uuuu uuuu
ADCBUF9	0292	—	—	—	—	—	—	ADC 数据缓冲器 9										0000 00uu uuuu uuuu
ADCBUFA	0294	—	—	—	—	—	—	ADC 数据缓冲器 10										0000 00uu uuuu uuuu
ADCBUFB	0296	—	—	—	—	—	—	ADC 数据缓冲器 11										0000 00uu uuuu uuuu
ADCBUFC	0298	—	—	—	—	—	—	ADC 数据缓冲器 12										0000 00uu uuuu uuuu
ADCBUFD	029A	—	—	—	—	—	—	ADC 数据缓冲器 13										0000 00uu uuuu uuuu
ADCBUFE	029C	—	—	—	—	—	—	ADC 数据缓冲器 14										0000 00uu uuuu uuuu
ADCBUFF	029E	—	—	—	—	—	—	ADC 数据缓冲器 15										0000 00uu uuuu uuuu
ADCON1	02A0	ADON	—	ADSIDL	—	—	—	FORM<1:0>		SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE	0000 0000 0000 0000
ADCON2	02A2	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>				BUFM	ALTS	0000 0000 0000 0000
ADCON3	02A4	—	—	—	SAMC<4:0>					ADRC	—	ADCS<5:0>						0000 0000 0000 0000
ADCHS	02A6	CH123NB<1:0>		CH123SB	CH0NB	CH0SB<3:0>				CH123NA<1:0>		CH123SA	CH0NA	CH0SA<3:0>				0000 0000 0000 0000
ADPCFG	02A8	—	—	—	—	—	—	—	—	—	—	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000 0000 0000 0000
ADCSSL	02AA	—	—	—	—	—	—	—	—	—	—	CSSL5	CSSL4	CSSL3	CSSL2	CSSL1	CSSL0	0000 0000 0000 0000

图注：　u = 未初始化位

注：　有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

19.0 系统集成

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

本系列器件的下列功能，旨在最大限度地提高系统可靠性，通过减少外部元件的使用来降低成本，提供省电工作模式以及提供代码保护：

- 振荡器选择
- 复位
  - 上电复位 （POR）
  - 上电延时定时器 （PWRT）
  - 振荡器起振定时器 （OST）
  - 可编程欠压复位 （BOR）
- 看门狗定时器 （WDT）
- 省电模式 （休眠和空闲）
- 代码保护
- 器件 ID 存储单元
- 在线串行编程 （ICSP）功能

dsPIC30F 器件具备一个看门狗定时器，它可以通过配置位永久使能，或用软件进行控制。它依靠自身的 RC 振荡器运行，以便提升可靠性。两个定时器提供了上电时必须的延时；一个是振荡器起振定时器 （OST），用来使芯片保持复位，直到晶体振荡器稳定为止；另一个是上电延时定时器 （PWRT），仅在上电时提供延时，用来在电源稳定过程中使器件保持在复位状态 。有了这两个片内定时器，大多数应用不再需要外部复位电路。

休眠模式，旨在提供极低电流的掉电模式。通过外部复位、看门狗定时器唤醒或中断，用户可将器件从休眠中唤醒。一些振荡器功能仍然可用，从而器件能够适用于范围宽广的各种应用。在空闲模式下，时钟源仍然处于工作状态，但 CPU 停止。RC 振荡器可降低系统成本，而 LP 晶振功能可降低功耗。

19.1 振荡器系统概述

dsPIC30F 振荡器系统包含以下模块和功能：

- 可选择多种外部和内部振荡器作为时钟源
- 片上 PLL，可提高内部工作频率
- 在各种时钟源之间进行切换的时钟切换机制
- 可编程时钟后分频器，可节省系统功耗
- 故障保护时钟监视器 （FSCM），可检测时钟故障并采取故障保护措施
- 时钟控制寄存器 （OSCCON）
- 用于主振荡器选择的配置位

表 19-1 汇总了 dsPIC30F 的振荡器工作模式。图 19-1 中给出了振荡器系统的简化框图。

配置位用于在上电复位 （POR）和欠压复位 （BOR）时确定时钟源。此后，可在许可的时钟源之间切换时钟源。 OSCCON 寄存器用于控制时钟切换并包含与系统时钟相关的状态位。

表 **19-1**： 振荡器工作模式

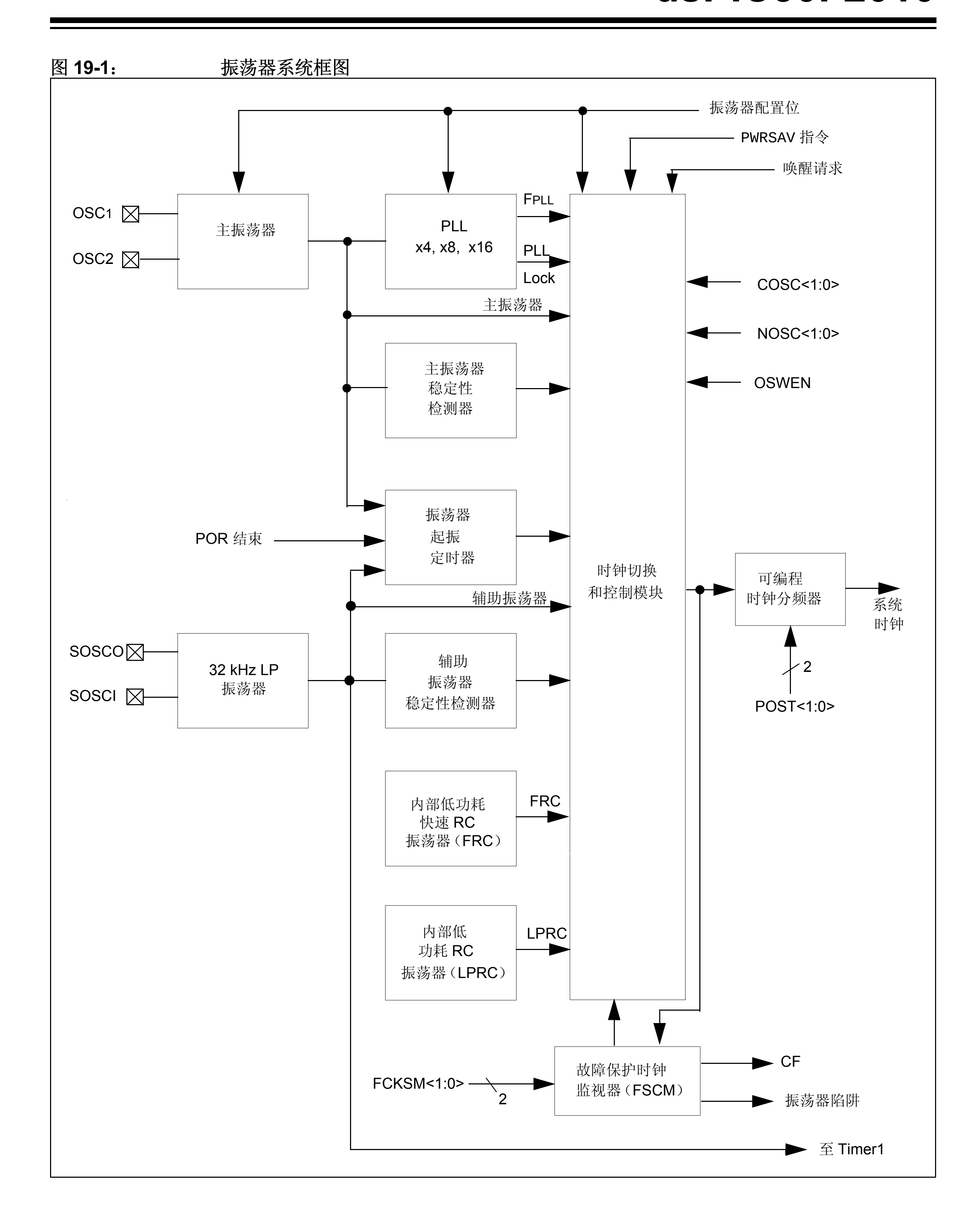
振荡器模式	描述
XTL	OSC1:OSC2 上 200 kHz-4 MHz 晶振。
XT	OSC1:OSC2 上 4 MHz-10 MHz 晶振。
XT w/ PLL 4x	OSC1:OSC2 上 4 MHz-10 MHz 晶振。 4x PLL 使能。
XT w/ PLL 8x	OSC1:OSC2 上 4 MHz-10 MHz 晶振。 8x PLL 使能。
XT w/ PLL 16x	OSC1:OSC2 上 4 MHz-10 MHz 晶振。16x PLL 使能 **(1)**。
LP	SOSCO:SOSCI 上 32 kHz 晶振 **(2)**。
HS	10 MHz-25 MHz 晶振。
EC	外部时钟输入 （0-40 MHz）。
ECIO	外部时钟输入 （0-40 MHz）。OSC2 引脚为 I/O。
EC w/ PLL 4x	外部时钟输入 （0-40 MHz）。OSC2 引脚为 I/O。 4x PLL 使能 **(1)**。
EC w/ PLL 8x	外部时钟输入 （0-40 MHz）。OSC2 引脚为 I/O。8x PLL 使能 **(1)**。
EC w/ PLL 16x	外部时钟输入 （0-40 MHz）。OSC2 引脚为 I/O。 16x PLL 使能 **(1)**。
ERC	外部 RC 振荡器。 OSC2 引脚为 Fosc/4 输出 **(3)**。
ERCIO	外部 RC 振荡器。 OSC2 引脚为 I/O**(3)**。
FRC	7.37 MHz 内部 RC 振荡器。
LPRC	512 kHz 内部 RC 振荡器。

注 **1**：必须满足 dsPIC30F 最大工作频率为 120 MHz 的要求。

2：LP 振荡器可方便地为系统时钟和 Timer1 实时时钟所共用。

3：需要外部 R、 C 元件。工作频率最高为 4 MHz。

图 **19-1：** 振荡器系统框图



19.2 振荡器配置

19.2.1 初始时钟源选择

当退出上电复位或欠压复位状态时，器件根据以下原则进行时钟源的选择：

a) 通过 FOS<1:0> 配置位选择四种振荡器组合中的一种。

b) 通过FPR<3:0>配置位选择主振荡器组中13个振荡器之一。

表 19-2 中显示了时钟选择。

19.2.2 振荡器起振定时器 （OST）

为确保晶振 （或陶瓷振荡器）已起振并达到稳定状态，振荡器中包含一个振荡器起振定时器。该定时器仅仅是一个 10 位计数器，在允许振荡器时钟应用于系统其他部分之前该计数器将计数 1024 个 TOSC 周期。超时周期称为 TOST。每当振荡器重启（即当 POR、BOR 和从休眠模式唤醒时）需进行 TOST 时间的定时。振荡器起振定时器适用于 LP 振荡器和主振荡器的 XT、XTL 和 HS 模式 （当 POR、 BOR 和从休眠模式唤醒时）。

表 19-2：　时钟选择的配置位值

振荡器模式	振荡器源	FOS1	FOS0	FPR3	FPR2	FPR1	FPR0	OSC2 功能
EC	主振荡器	1	1	1	0	1	1	CLKO
ECIO	主振荡器	1	1	1	1	0	0	I/O
EC w/ PLL 4x	主振荡器	1	1	1	1	0	1	I/O
EC w/ PLL 8x	主振荡器	1	1	1	1	1	0	I/O
EC w/ PLL 16x	主振荡器	1	1	1	1	1	1	I/O
ERC	主振荡器	1	1	1	0	0	1	CLKO
ERCIO	主振荡器	1	1	1	0	0	0	I/O
XT	主振荡器	1	1	0	1	0	0	OSC2
XT w/ PLL 4x	主振荡器	1	1	0	1	0	1	OSC2
XT w/ PLL 8x	主振荡器	1	1	0	1	1	0	OSC2
XT w/ PLL 16x	主振荡器	1	1	0	1	1	1	OSC2
XTL	主振荡器	1	1	0	0	0	X	OSC2
HS	主振荡器	1	1	0	0	1	X	OSC2
LP	辅助振荡器	0	0	—	—	—	—	（注 **1, 2**）
FRC	内部 FRC	0	1	—	—	—	—	（注 **1, 2**）
LPRC	内部 LPRC	1	0	—	—	—	—	（注 **1, 2**）

注　**1：** OSC2 引脚功能由主振荡器模式选择确定 （FPR<3:0>）。

　　2： 注意，即使始终选择辅助振荡器或内部时钟源， OSC1 引脚仍不能用作 I/O 引脚。

19.2.3 LP 振荡器控制

通过以下各位可使能 LP 振荡器：

1. 当前振荡器组控制位 COSC<1:0>。
2. LPOSCEN 位 （OSCCON 寄存器）。

如果 LPOSCEN = 1，则 LP 振荡器被使能 （即使处于休眠模式）。在下列条件下，LP振荡器将作为器件时钟：

- COSC<1:0> = 00 （选择 LP 作为主振荡器）且
- LPOSCEN = 1

保持 LP 振荡器总是使能可以快速切换至 32kHz 系统时钟以实现低功耗运行。返回更快速的主振荡器仍需要经过起振时间。

19.2.4 锁相环 （PLL）

PLL 可将主振荡器或快速 RC 振荡器产生的时钟信号进行频。 PLL 可选择具有 x4、 x8 和 x16 的增益。表 19-3 中对输入和输出频率范围进行了汇总。

表 19-3： PLL 频率范围

FIN	PLL 倍频	FOUT
4 MHz-10 MHz	x4	16 MHz-40 MHz
4 MHz-10 MHz	x8	32 MHz-80 MHz
4 MHz-7.5 MHz	x16	64 MHz-120 MHz

PLL 具有锁定输出，当 PLL 进入锁相状态时，锁定输出有效。如果锁相环锁定失败 （例如由噪声导致），锁定信号将为无效。此信号的状态将通过 OSCCON 寄存器中的只读位 LOCK 反映。

19.2.5 快速 RC 振荡器 （FAST RC OSCILLATOR， FRC）

FRC 振荡器是一个快速（7.37 MHz ±2% 标称值）的内部 RC 振荡器。该振荡器用于提供合理的器件工作速度而无需使用外部晶振、陶瓷振荡器或 RC 网络。

当 OSCCON 寄存器中的振荡器选择控制位（OSCCON<13:12>）设定为 ‘01’时， dsPIC30F 将使用 FRC 振荡器进行工作。

通过 TUN<3:0>（OSCCON<15:14> 和 OSCCON<11:10>）指定的 4 位位域将使用户可对内部快速 RC 振荡器 （标称值为 7.37 MHz）进行调节。用户可在厂商校准设定值的 -12%（-960 kHz）至 +10.5%（+840 kHz）范围内对 FRC 振荡器进行调节，单步调节的幅度为 1.50%，参见表 19-4。

表 19-4： FRC 调节

TUN<3:0> 位	FRC 频率
0111	+ 10.5%
0110	+ 9.0%
0101	+ 7.5%
0100	+ 6.0%
0011	+ 4.5%
0010	+ 3.0%
0001	+ 1.5%
0000	中心频率 （振荡器运行于校准频率）
1111	- 1.5%
1110	- 3.0%
1101	- 4.5%
1100	- 6.0%
1011	- 7.5%
1010	- 9.0%
1001	- 10.5%
1000	- 12.0%

19.2.6 低功耗 RC 振荡器 （LPRC）

LPRC 振荡器是看门狗定时器（WDT）的组成部分，其标称振荡频率为 512 kHz。 LPRC 振荡器是上电延时定时器 （PWRT）电路、 WDT 和时钟监视器电路的时钟源。它也可用于低功耗要求高但时序精度要求不高的应用场合作为低频时钟源。

由于 LPRC 振荡器是 PWRT 的时钟源，因此上电复位时该振荡器总是被使能。当 PWRT 超时时，如果以下任何一个条件为真， LPRC 振荡器将保持使能状态：

- 故障保护时钟监视器使能
- WDT 使能
- 通过OSCCON寄存器中的控制位COSC<1:0>选择 LPRC 振荡器作为系统时钟

如果以上任何一个条件都不为真，在 PWRT 超时之后 LPRC 将被关闭。

> **注 1：** 通过主振荡器模式选择位 （FPR<3:0>）可确定 OSC2 的引脚功能。
>
> **2：** 即使总是选择辅助振荡器或内部时钟源， OSC1 引脚也不能用作 I/O 引脚。

19.2.7 故障保护时钟监视器

故障保护时钟监视器 （FSCM），使得器件能够继续工作，即便是振荡器出现故障。恰当地设置 FOSC 器件配置寄存器中的 FCKSM 配置位（时钟切换和监视器选择位），可使能 FSCM 功能。如果使能了 FSCM 功能，LPRC 内部振荡器将始终保运行 （休眠模式除外），不再受 SWDTEN 位的控制。

在发生振荡器故障时， FSCM 会产生时钟故障陷阱事件，并将系统时钟切换到 FRC 振荡器。之后，用户即可选择尝试重新启动振荡器，或者执行受控关闭操作。用户可以把陷阱当作热复位来处理，这只要把复位地址载入振荡器陷阱向量即可。出现故障时，一旦识别出时钟故障， CF （时钟故障）状态位 （OSCCON<3>）也将置 1。

出现时钟故障时， WDT 不受影响，继续靠 LPRC 时钟运行。

从 POR、 BOR 或休眠退出后，如果振荡器起振时间非常慢的话，则可能出现这样的情况，在振荡器启动之前 PWRT 定时器就已到期。在这样的情形下，FSCM 将被激活，FSCM 将启动一个时钟故障陷阱，FRC 振荡器选择将载入 COSC<1:0> 位中。这实际上是关闭了先前正在试图启动的振荡器。

在时钟故障陷阱 ISR 中，用户可以检测到这样的情况并重启振荡器。

检测到时钟故障时， FSCM 模块将发起至 FRC 振荡器的时钟切换，方法如下：

1. 把 FRC 振荡器选择位的值载入 COSC 位（OSCCON<13:12>）中。
2. CF 位 （OSCCON<3>）置 1。
3. OSWEN 控制位 （OSCCON<0>）清零。

出于时钟切换的目的，时钟源划分为四组：

1. 主组
2. 辅助组
3. 内部 FRC 组
4. 内部 LPRC 组

用户可以在这些功能组之间进行切换，但不能在同一个组内进行切换。如果选择了主组的话，则在主组内的选择始终由 FPR<3:0> 配置位决定。

与时钟切换相关的控制和状态位在OSCCON寄存器中：

- COSC<1:0>：只读状态位，总是表明当前有效的振荡器组。
- NOSC<1:0>：控制位，写控制位表明选择了新的振荡器组。
  - 在POR和BOR时，将把配置位FOS<1:0>的值装入 COSC<1:0> 和 NOSC<1:0> 中。
- LOCK：LOCK 状态位，表明 PLL 是否锁定。
- CF：只读状态位，说明是否检测到时钟故障。
- OSWEN：控制位，当时钟切换过程启动时，从 0 变为 1。清零 OSWEN 控制位将中止正在进行中的时钟切换 （用于挂起情形）。

如果配置位 FCKSM<1:0> = 1x，那么时钟切换和故障保护时钟监视器功能被禁止。这是缺省的配置位设定。

如果时钟切换禁止，则 FOS<1:0> 和 FPR<3:0> 位直接控制振荡器选择， COSC<1:0> 位不控制时钟选择。然而，这些位将反映时钟源选择。

注：	当使能故障保护时钟监视器时，在应用中不应尝试切换至低于100 KHz的时钟频率。如果执行此了这样的时钟切换，器件可能产生振荡器故障陷阱并切换至快速RC振荡器。

19.2.8 防止意外写入 OSCCON

由于 OSCCON 寄存器控制时钟切换和时钟分频，因此有意地将其写入操作过程设计得较为困难。

要写入 OSCCON 低字节，必须执行以下代码序列且不应在其中插入任何其他指令：

- 写入字节 “0x46”至 OSCCON 低字节
- 写入字节 “0x57”至 OSCCON 低字节

允许在一个指令周期内对OSCCONL进行字节写操作。 写入期望值或使用位操作指令。

要写入 OSCCON 高字节，必须执行以下指令且不应在其中插入任何其他指令：

- 写入字节 “0x78”至 OSCCON 高字节
- 写入字节 “0x9A”至 OSCCON 高字节

允许在一个指令周期内对OSCCONL进行字节写操作。 写入期望值或使用位操作指令。

19.3 复位

dsPIC30F2010 区分下列各种复位：

a) 上电复位（POR）
b) 正常工作期间的 $\overline{\text{MCLR}}$ 复位
c) 休眠期间的 $\overline{\text{MCLR}}$ 复位
d) 看门狗定时器（WDT）复位（在正常工作期间）
e) 可编程欠压复位（BOR）
f) RESET 指令
g) 由陷阱锁定（TRAPR）导致的复位
h) 由非法操作码导致的复位，或者是由于把未初始化的 W 寄存器用作地址指针（IOPUWR）而导致的复位

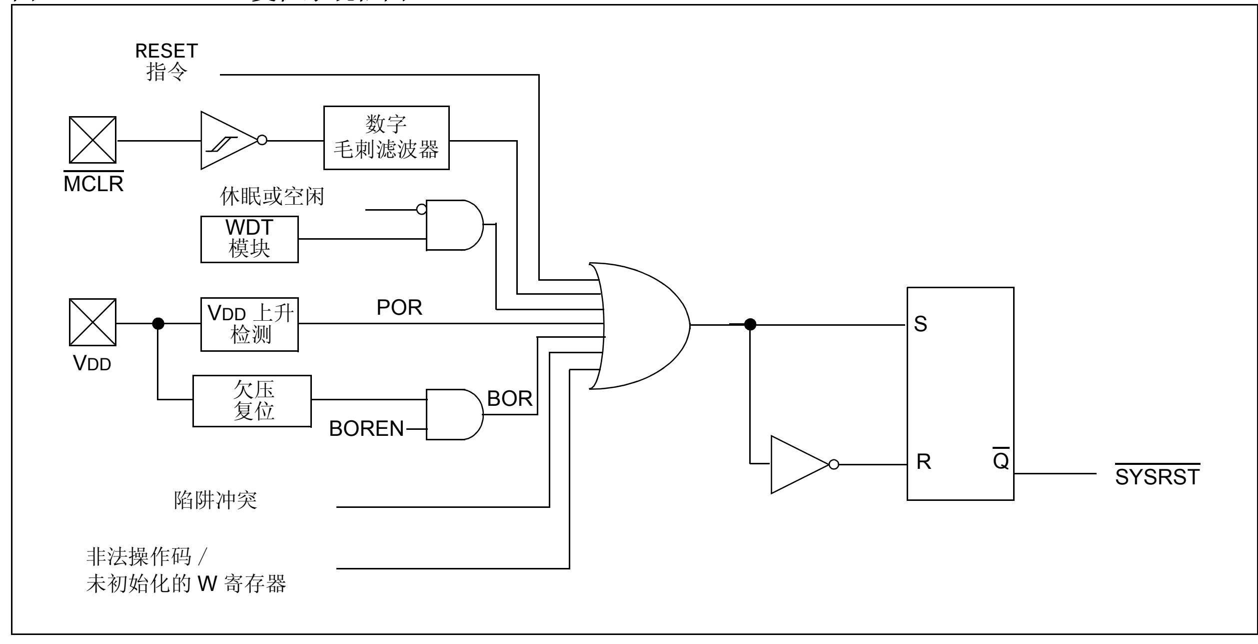
各种复位条件以不同的方式影响不同的寄存器。大多数寄存器不受 WDT 唤醒的影响，因为这被视为是正常工作的继续。在不同的复位条件下，将以不同的方式置 1 或清零 RCON 寄存器中的状态位，如表 19-5 所示。软件中使用这些位来确定复位的性质。

片内复位电路的框图，如图 19-2 所示。

在 $\overline{\text{MCLR}}$ 复位路径上，提供了一个 $\overline{\text{MCLR}}$ 噪声滤波器。滤波器检测并滤除小脉冲。

内部产生的复位不会将 $\overline{\text{MCLR}}$ 引脚驱动为低电平。

图 19-2： 复位系统框图



19.3.1 POR：上电复位

当检测到 VDD 上升时，上电事件将产生内部 POR 脉冲。复位脉冲将在 POR 电路阈值电压（VPOR）处产生，标称阈值是 1.85V。器件电源电压特性曲线必须满足规定的起始电压和上升速度要求。POR 脉冲将复位 POR 定时器，并使器件进入复位状态。POR 还将选择器件时钟源，时钟源由振荡器配置位指定。

POR 电路将插入一个标称值为 10 µs 的小延时 TPOR，确保器件偏压电路是稳定的。此外，还要施加用户选择的上电延时（TPWRT）。TPWRT 参数由器件配置位设定，可以是 0 ms（无延时）、4 ms、16 ms 或 64 ms。器件上电的总延时为 TPOR + TPWRT。在这些延时结束后，$\overline{\text{SYSRST}}$ 将在 Q1 时钟的下一个上升沿反相，PC 将跳转到复位向量。

$\overline{\text{SYSRST}}$ 信号时序如图 19-3 至图 19-5 所示。

图 19-3： 上电过程中的延时时序（MCLR 接至 VDD）

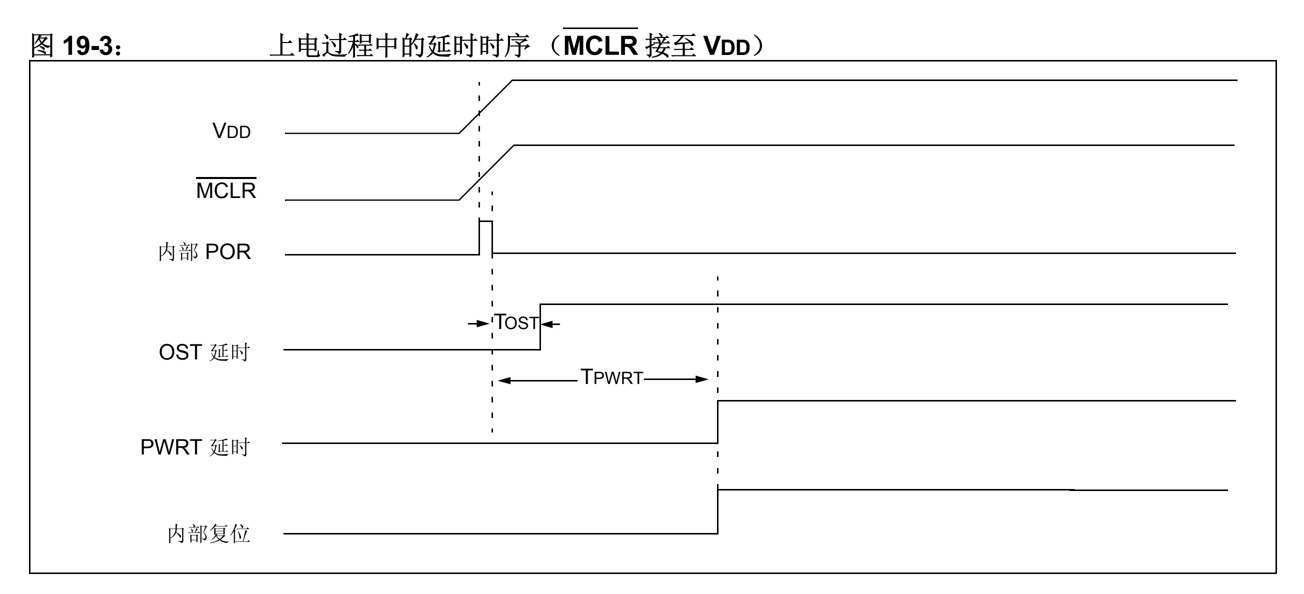


图 19-4： 上电过程中的延时时序（MCLR 未接至 VDD）：情形 1

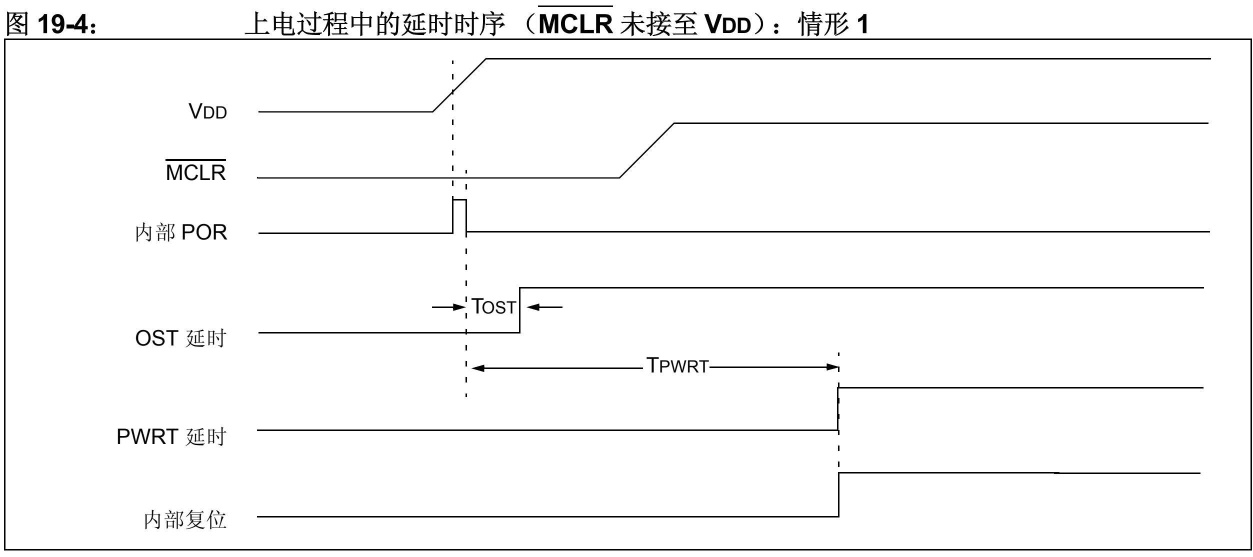
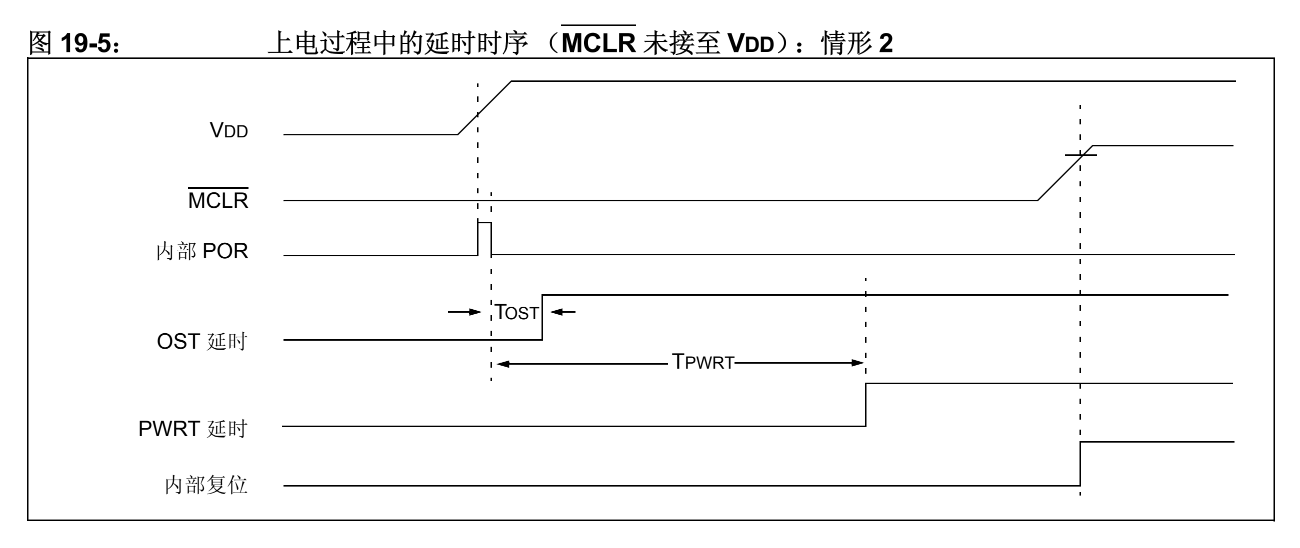


图 19-5： 上电过程中的延时时序（MCLR 未接至 VDD）：情形 2



19.3.1.1 具有长晶振起振时间的 POR （FSCM 使能）

振荡器起振电路没有连接到 POR 电路。一些晶振电路（尤其是低频晶振）具有相对长的起振时间。从而，在 POR 定时器和 PWRT 超时后，可能会出现下列情况：

- 振荡器电路还没有开始振荡。
- 振荡器起振定时器还没有超时 （如果使用晶振的话）。
- PLL 还未锁定 （如果使用了 PLL 的话）。

如果 FSCM 使能且出现上述条件之一时，就将产生时钟故障陷阱。器件将自动切换到 FRC 振荡器，而用户可在陷阱 ISR 中切换到期望的晶体振荡器。

19.3.1.2 FSCM 和 PWRT 禁止时的工作

如果 FSCM 禁止、且上电延时定时器（PWRT）也禁止的话，上电时器件将快速退出复位状态。如果时钟源是 FRC、 LPRC、 ERC 或 EC，它将立即激活。

如果 FSCM 禁止而系统时钟还没有启动，器件将在复位向量处处于冻结状态，直到系统时钟启动为止。从用户的角度看，器件看上去仍然在复位状态，直到系统时钟可用为止。

19.3.2 BOR：可编程欠压复位

BOR （欠压复位）模块是基于内部参考电压电路的。BOR 模块的主要用途是在发生欠压条件时产生器件复位。欠压条件通常由 AC 电源线上的干扰信号 （即，由于电源传输线路不良造成的 AC 周期波形丢失，或者由于接入大负载时过电流造成电压下降）产生的。

BOR 模块允许选择以下电压跳变点之一：

- 2.6V-2.71V
- 4.1V-4.4V
- 4.58V-4.73V

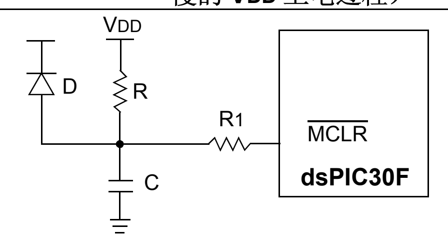
注：	这里的 BOR 电压跳变点为标称值，仅供设计参考。

BOR 将产生复位器件的复位脉冲。在欠压复位时，会根据器件配置位（FOS<1:0> 和 FPR<3:0>）选择时钟源。此外，如果选择了晶体振荡器模式，BOR 将激活振荡器起振定时器 （OST）。系统时钟将保持到 OST 超时。如果使用了 PLL，则时钟将被保持到 LOCK 位（OSCCON<5>）置 1。

同时，在内部复位信号释放之前，还要施加 POR 延时（TPOR）和 PWRT 延时 （TPWRT）。如果 TPWRT = 0 且正在使用的是晶体振荡器的话，那么将施加 TFSCM = 100 μs 的标称延时。这样，总延时为 TPOR + TFSCM。

BOR 状态位（RCON<1>）将置1，以表明发生了BOR。使能时，BOR 电路将在休眠或空闲模式下继续工作，当 VDD 下降到 BOR 阈值电压以下时将复位器件。

图 19-6： 外部上电复位电路 （对于较慢的 VDD 上电过程）



注 1： 只有当 VDD 上电速率太慢时，才需要外部上电复位电路。二极管 D 有助于当 VDD 断电时对电容进行快速放电。

2： 应该选择合适的 R 以确保 R 两端的压降不会违反器件电气规范。

3： 应该选择合适的 R1 以限制静电放电（Electrostatic Discharge，ESD）或过度电性应力（Electrical Overstress，EOS）导致 MCLR/VPP 引脚损坏时从外部电容 C 流入 MCLR 的电流。

注：	专用监视器件，如 MCP1XX 和 MCP8XX 也可用作外部上电复位电路。

表 19-5 所示为 RCON 寄存器的复位状态。由于 RCON 寄存器中的控制位都是可读写的，表中的信息意味着：所有位状态是在“条件”这一列指定操作之前的位状态的反码。

表 19-5： RCON 寄存器的初始状态，情形 1

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	空闲	休眠	POR	BOR
上电复位	0x000000	0	0	0	0	0	0	0	1	1
欠压复位	0x000000	0	0	0	0	0	0	0	0	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	0	0	0
正常工作期间的软件复位	0x000000	0	0	0	1	0	0	0	0	0
休眠模式下的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	1	0	0
空闲模式下的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	1	0	0	0
WDT 超时复位	0x000000	0	0	0	0	1	0	0	0	0
WDT 唤醒	PC + 2	0	0	0	0	1	0	1	0	0
被中断从休眠状态唤醒	PC + 2[1]	0	0	0	0	0	0	1	0	0
时钟故障陷阱	0x000004	0	0	0	0	0	0	0	0	0
陷阱复位	0x000000	1	0	0	0	0	0	0	0	0
非法操作陷阱	0x000000	0	1	0	0	0	0	0	0	0

图注： u = 不变， x = 未知， - = 未用位，读为 0

注 1： 如果唤醒由允许的中断导致， PC 将装入相应的中断向量。

表 19-6 显示了 RCON 寄存器位状态的第二个实例。在该情形中，并未假定在条件栏指定的操作之前用户已置 1/ 清零特定的位。

表 19-6： RCON 寄存器的初始状态，情形 2

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	IDLE	SLEEP	POR	BOR
上电复位	0x000000	0	0	0	0	0	0	0	1	1
欠压复位	0x000000	u	u	u	u	u	u	u	0	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	0	0	0	0	u	u
正常工作期间的软件复位	0x000000	u	u	0	1	0	0	0	u	u
休眠模式下的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	0	1	u	u
空闲模式下的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	1	0	u	u
WDT 超时复位	0x000000	u	u	0	0	1	0	0	u	u
WDT 唤醒	PC + 2	u	u	u	u	1	u	1	u	u
被中断从休眠状态唤醒	PC + 2 [1]	u	u	u	u	u	u	1	u	u
时钟故障陷阱	0x000004	u	u	u	u	u	u	u	u	u
陷阱复位	0x000000	1	u	u	u	u	u	u	u	u
非法操作陷阱	0x000000	u	1	u	u	u	u	u	u	u

图注： u = 不变， x = 未知， - = 未用位，读为 0

注 1： 如果唤醒由允许的中断导致， PC 将装入相应的中断向量。

19.4 看门狗定时器 （WDT）

19.4.1 看门狗定时器的工作

看门狗定时器 （WDT）的主要功能是在软件故障时使处理器复位。 WDT 是独立运行的定时器，它使用片内 RC 振荡器，不需要外部元件。于是，即使主处理器时钟（例如，晶振）出现故障，WDT 定时器将继续工作。

19.4.2 使能和禁止 WDT

看门狗定时器的使能或禁止，只能通过配置寄存器 FWDT 中的配置位 （FWDTEN）来进行。

设置 FWDTEN = 1，使能看门狗定时器。使能是在对器件进行编程时完成的。缺省情况下，在擦除芯片后，FWDTEN 位 = 1。任何能够对 dsPIC30F 器件进行编程的编程器，都能够对这个配置位及其他配置位进行编程。

如果使能的话，WDT 将递增，直到它溢出或"超时"。WDT 超时将强制器件复位 （除非是在休眠期间）。要防止 WDT 超时，用户必须使用 CLRWDT 指令清零看门狗定时器。

如果休眠期间 WDT 超时，器件将唤醒。RCON 寄存器中的 WDTO 位将清零，表明 WDT 超时导致唤醒。

设置 FWDTEN = 0，允许用户软件通过 SWDTEN 控制位 （RCON<5>）来使能 / 禁止看门狗定时器。

19.5 省电模式

通过执行一条特殊的指令 PWRSAV，可以进入两个省电模式。

这两个模式是：休眠和空闲。

PWRSAV 指令的格式如下：

PWRSAV <parameter>，其中"parameter"定义是空闲还是休眠模式。

19.5.1 休眠模式

在休眠模式下，CPU 和外设的时钟关闭。如果在使用片内振荡器，它也将关闭。

休眠期间，故障保护时钟监视器不起作用，因为没有时钟需要监视。不过，如果休眠期间 WDT 工作的话，LPRC 时钟将保持为活动状态。

欠压保护电路和低电压检测电路，如果使能的话，在休眠期间仍然起作用。

发生以下任一事件时，处理器将从休眠模式唤醒：

- 任何中断，如果已使能并且满足所需的优先级
- 任何复位 （POR、 BOR 或 $\overline{\text{MCLR}}$）
- WDT 超时

从休眠模式唤醒时，处理器将重新启动进入休眠模式之前处于活动状态的时钟。当时钟切换使能时，COSC<1:0> 位将决定唤醒时所使用的振荡器源。如果时钟切换禁止，则只有一个系统时钟。

注：	如果发生 POR 或 BOR，则振荡器的选择将基于 FOS<1:0> 和 FPR<3:0> 配置位。

如果时钟源是振荡器，器件的时钟将一直关闭，直到 OST 超时为止（这表明振荡器已经稳定）。如果使用了 PLL，系统时钟关闭，直到 LOCK = 1 为止（这表明 PLL 已经稳定）。不管是哪种情况，都要加上 TPOR、TLOCK 和 TPWRT 延时。

如果使用了 EC、FRC、LPRC 或 ERC 振荡器，那么要加上 TPOR 的延时 （~ 10 µs）。这是从休眠中唤醒时的最小延时。

此外，如果在休眠期间 LP 振荡器处于工作状态，并且 LP 是唤醒时使用的振荡器，那么起振延时等于 TPOR。不会有 PWRT 延时和 OST 定时器延时。为了使从休眠中唤醒的起振延时最小，在进入休眠之前，应该选择这些快速唤醒选项之一。

任何中断，如果被单独允许 （通过相应的 IE 位）并且优先级占先的话，将唤醒处理器。处理器将处理中断，转移至ISR。唤醒时RCON寄存器中的休眠状态位将置1。

注：	尽管加入各种不同的延迟时间 （TPOR、TLOCK 和 TPWRT），在延时结束时晶振（和 PLL）可能并未处于工作状态 （例如，对于低频晶振）。在这种情况下，如果 FSCM 被使能，则器件将这这种情况作为时钟故障并处理时钟故障陷阱， FRC 振荡器将被使能而用户必须重新使能晶振。如果 FSCM 未使能，器件将暂停代码执行直至时钟稳定且将继续处于休眠模式直至振荡器时钟启动。

所有复位将使处理器从休眠模式唤醒。除 POR 之外的任何复位都将置 1 休眠状态位。在 POR 时，休眠状态位将被清零。

如果看门狗定时器被使能，则处理器将在发生 WDT 超时时从休眠模式唤醒。休眠和 WDTO 状态位都将被置 1。

19.5.2 空闲模式

在空闲模式下，CPU 的时钟关闭，而外设继续运行。与休眠模式不同的是，时钟源继续保持为工作状态。

对于一些外设而言，每个模块中有一个与外设对应的控制位，控制外设在空闲期间是否工作。

如果时钟故障检测使能的话，LPRC 故障保护时钟继续保持为工作状态。

在发生以下事件之一时，处理器将从空闲模式唤醒：

- 任何中断，如果已允许（IE 为 1）并且满足所需的优先级
- 任何复位 （POR、 BOR 或 $\overline{\text{MCLR}}$）
- WDT 超时

在从空闲模式唤醒时，时钟再次供 CPU 使用且指令立即开始执行，从 PWRSAV 指令之后的第一条指令处开始执行。

任何中断，如果已允许 （使用相应的 IE 位）并且优先级占先的话，将唤醒处理器。处理器将处理中断，转移至 ISR。唤醒时 RCON 寄存器中的空闲状态位将置 1。

除 POR 之外的任何复位，都将把空闲状态位置 1。POR 时，空闲状态位清零。

如果看门狗定时器使能，则处理器在 WDT 超时时将从空闲模式中唤醒。空闲和 WDTO 状态位都将置 1。

与从休眠中唤醒不同的，从空闲中唤醒时不存在延时。

19.6 器件配置寄存器

每个器件配置寄存器中的配置位都指定某些器件模式，配置位通过器件编程器或者器件的在线串行编程（ICSP）功能来进行编程。虽然每个器件配置寄存器都是 24 位寄存器，但只有低 16 位可用来保存配置数据。有四个器件配置寄存器可供用户使用：

1. FOSC（0xF80000）：振荡器配置寄存器
2. FWDT（0xF80002）：看门狗定时器配置寄存器
3. FBORPOR（0xF80004）：BOR 和 POR 配置寄存器
4. FGS （0xF8000A）：通用代码段配置寄存器

如果在器件编程器中选择了器件，配置位的位置将被自动处理。可以在源代码（这取决于使用的语言工具）中，或者通过编程接口来指定配置位的期望状态。在对器件编程之后，应用软件可以通过表读指令读取配置位的值。更多信息，请参见器件的编程规范。

注：	如果设置了代码保护配置控制位（FGS<GCP> 和 FGS<GWRP>），那么只有在电压 VDD ≥ 4.5V 时才可以擦除整个代码保护的器件。

19.7 在线调试器

如果选择 MPLAB® ICD 2 作为调试器，则使能在线调试功能。与 MPLAB IDE 一起使用时，在线调试功能允许简单的调试功能。当使能器件的这个功能时，某些资源不能用于通常的用途。这些资源包括数据 RAM 的前 80 字节 和两个 I/O 引脚。

用户使用 MPLAB IDE 中的配置选项，可选择四对调试 I/O引脚中的一对。这些引脚为EMUD/EMUC、EMUD1/EMUC1、 EMUD2/EMUC2 和 EMUD3/EMUC3。

无论选取哪对调试引脚，选取的 EMUD 引脚都是仿真 / 调试数据线，而 EMUC 引脚则是仿真 / 调试时钟线。这些引脚将与 Microchip 提供的 MPLAB ICD 2 模块接口。MPLAB ICD 2 使用选取的调试 I/O 引脚，发送命令、接收响应，发送和接收数据。要使用器件的在线调试功能，应用设计必须实现至 MCLR、VDD、VSS、PGC、PGD 以及选取的EMUDx/EMUCx引脚对的ICSP 连接。

这样就产生了两种可能性：

1. 如果选取 EMUD/EMUC 作为调试 I/O 引脚对，那么只需要一个 5 引脚的接口，这是因为，在所有的 dsPIC30F 器件上， EMUD 和 EMUC 引脚功能均可与 PGD 和 PGC 引脚功能复用。
2. 如果选取了 EMUD1/EMUC1、 EMUD2/EMUC2 或 EMUD3/EMUC3 作为调试 I/O 引脚对，那么需要 7 引脚的接口，这是因为，EMUDx/EMUCx 引脚功能（x = 1、2 或 3）不能与 PGD 和 PGC 引脚功能复用。

表 **19-7：** 系统集成寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
RCON	0740	TRAPR	IOPUWR	BGST	—	—	—	—	—	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	取决于复位类型。
OSCCON	0742	TUN3	TUN2	COSC<1:0>		TUN1	TUN0	NOSC<1:0>		POST<1:0>		LOCK	—	CF	—	LPOSCEN	OSWEN	取决于配置位。

图注： u = 未初始化位

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

表 **19-8：** 器件配置寄存器映射

文件寄存器名称	地址	bit 23-16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FOSC	F80000	—	FCKSM<1:0>		—	—	—	—	FOS<1:0>		—	—	—	—	FPR<3:0>			
FWDT	F80002	—	FWDTEN	—	—	—	—	—	—	—	—	—	FWPSA<1:0>		FWPSB<3:0>			
FBORPOR	F80004	—	MCLREN	—	—	—	—	PWMPIN	HPOL	LPOL	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>	
FGS	F8000A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GCP	GWRP

注： 有关寄存器各位的描述，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

注：

20.0 指令集汇总

> 注：本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》（DS70046E_CN）。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

dsPIC30F 指令集的众多增强性能扩展了以往的 PIC® MCU 指令集，同时可以方便地从 PIC MCU 指令集移植到 dsPIC30F 指令集。

大部分指令的长度为一个程序存储字（24 位）。只有三条指令需要两个程序存储单元。

每一条单字指令长 24 位，分为一个指定指令类型的 8 位操作码和进一步指定指令操作的一个或多个操作数。

指令集是高度正交的，分为五个基本类别：

- 针对字或字节的操作
- 针对位的操作
- 立即数操作
- DSP 操作
- 控制操作

表 20-1 给出了在说明指令时使用的通用符号。

表 20-2 是dsPIC30F 指令集汇总，还给出了每条指令影响的状态标志位。

大多数针对字或字节的 W 寄存器指令（包括桶形移位指令）有三个操作数：

- 第一个源操作数通常是寄存器 Wb，不带任何地址修改量
- 第二个源操作数通常是寄存器 Ws，带或不带地址修改量
- 运算结果的目的寄存器，通常是寄存器 Wd，带或不带地址修改量

不过，针对字或字节的文件寄存器指令只有两个操作数：

- 文件寄存器由 f 值指定
- 目的寄存器可以是文件寄存器 f 或 W0 寄存器（用 WREG 表示）

大多数位操作类指令（包括简单的循环 / 移位指令）有两个操作数：

- W 寄存器（带或不带地址修改量）或文件寄存器（由 Ws 或 f 的值指定）
- W 寄存器或文件寄存器中的位（由一个立即数指定，或者由 Wb 寄存器的内容间接指定）

涉及数据传送的立即数指令，可以使用下列操作数：

- 要被装入到 W 寄存器或文件寄存器中的立即数（由 k 值指定）
- 要装入立即数的W寄存器或文件寄存器（由Wb或f 指定）

然而，涉及算术或逻辑操作的立即数指令，使用如下的操作数：

- 第一个源操作数是寄存器 Wb，不带任何地址修改量
- 第二个源操作数是立即数
- 操作结果的目的寄存器（仅在与第一个源操作数不同时）通常是寄存器 Wd（带或不带地址修改量）

`MAC` 类 DSP 指令可使用下列操作数：

- 要使用的累加器（A 或 B）（必须的操作数）
- 要用作第二个操作数的 W 寄存器
- X 和 Y 地址空间预取操作
- X 和 Y 地址空间预取目的寄存器
- 累加器回写目的寄存器

与乘法无关的其他 DSP 指令使用的操作数可能包括：

- 要使用的累加器（必须）
- 源操作数或目的操作数（分别由 Wso 或 Wdo 指定），带或不带地址修改量
- 移位位数，由 W 寄存器 Wn 或立即数指定

控制指令可以使用下列操作数：

- 程序存储地址
- 表读和表写指令的模式

除了某些双字指令外，所有指令都是单字指令；双字指令之所以是双字长的（48 位），是因为要用 48 位来提供所需信息。在第二个字中，8 个 MSb 全为 `0`。如果指令自身把第二个字当作一条指令来执行的话，它将作为一条 `NOP` 指令来执行。

大多数单字长指令都在一个指令周期内执行，除非条件测试为真、或者指令运行结果要改变程序计数器。在这样的情形下，执行就需要两个指令周期，多出的一个指令周期，就如同执行了一条 NOP 指令。值得注意的例外是 BRA（无条件 / 计算转移）、间接 CALL/GOTO、所有的表读和表写以及 RETURN/RETFIE 指令，它们是单字长指令，但执行需要两个或三个周期。某些与跳过后续指令有关的指令，如果要执行跳过的话，可能需要两个或三个周期，这取决于被跳过的指令是单字还是双字指令。此外，双字传送需要两个周期。双字指令执行需要两个指令周期。

注： 关于指令集的更多细节，可参见《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

表 20-1： 操作码描述中使用的符号

字段	描述
#text	表示由“text”定义的立即数
(text)	表示“text 的内容”
[text]	表示“text 寻址的地址单元”
{ }	可选的位域或操作
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.S	影子寄存器选择
.w	字模式选择 （默认）
Acc	两个累加器 {A, B} 之一
AWB	累加器回写目的地址寄存器 ∈ {W13, [W13] + = 2}
bit4	4 位位选择字段 （用于字寻址指令） ∈ {0...15}
C, DC, N, OV, Z	MCU 状态位：进位、半进位、负、溢出和全零位
Expr	绝对地址、标号或表达式 （由链接器解析）
f	文件寄存器地址 ∈ {0x0000...0x1FFF}
lit1	1 位无符号立即数 ∈ {0,1}
lit4	4 位无符号立即数 ∈ {0...15}
lit5	5 位无符号立即数 ∈ {0...31}
lit8	8 位无符号立即数 ∈ {0...255}
lit10	10 位无符号立即数，对于字节模式， ∈ {0...255}；对于字模式， ∈{0:1023}
lit14	14 位无符号立即数 ∈ {0...16384}
lit16	16 位无符号立即数 ∈ {0...65535}
lit23	23 位无符号立即数 ∈ {0...8388608}； LSB 必须为 0
None	字段无需内容，可为空
OA, OB, SA, SB	DSP 状态位：ACCA 溢出、 ACCB 溢出、 ACCA 饱和和 ACCB 饱和
PC	程序计数器
Slit10	10 位有符号立即数 ∈ {-512...511}
Slit16	16 位有符号立即数 ∈ {-32768...32767}
Slit6	6 位有符号立即数 ∈ {-16...16}

表 **20-1**： 操作码描述中使用的符号 （续）

字段	描述
Wb	基准 W 寄存器 ∈ {W0..W15}
Wd	目的 W 寄存器 ∈ { Wd, [Wd], [Wd++], [Wd--], [++Wd], [--Wd] }
Wdo	目的 W 寄存器 ∈ { Wnd, [Wnd], [Wnd++], [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb] }
Wm,Wn	被除数和除数工作寄存器对 （直接寻址）
Wm*Wm	用于平方指令的被乘数和乘数寄存器对 ∈ {W4 * W4,W5 * W5,W6 * W6,W7 * W7}
Wm*Wn	用于 DSP 指令的被乘数和乘数寄存器对 ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}
Wn	16 个工作寄存器之一 ∈ {W0..W15}
Wnd	16 个目的工作寄存器之一 ∈ {W0..W15}
Wns	16 个源工作寄存器之一 ∈ {W0..W15}
WREG	W0 （文件寄存器指令中使用的工作寄存器）
Ws	源 W 寄存器 ∈ { Ws, [Ws], [Ws++], [Ws--], [++Ws], [--Ws] }
Wso	源 W 寄存器 ∈ { Wns, [Wns], [Wns++], [Wns--], [++Wns], [--Wns], [Wns+Wb] }
Wx	用于 DSP 指令的 X 数据空间预取地址寄存器 ∈ {[W8] + = 6, [W8] + = 4, [W8] + = 2, [W8], [W8] - = 6, [W8] - = 4, [W8] - = 2, [W9] + = 6, [W9] + = 4, [W9] + = 2, [W9], [W9] - = 6, [W9] - = 4, [W9] - = 2, [W9 + W12], 无 }
Wxd	用于 DSP 指令的 X 数据空间预取目的寄存器 ∈ {W4..W7}
Wy	用于 DSP 指令的 Y 数据空间预取地址寄存器 ∈ {[W10] + = 6, [W10] + = 4, [W10] + = 2, [W10], [W10] - = 6, [W10] - = 4, [W10] - = 2, [W11] + = 6, [W11] + = 4, [W11] + = 2, [W11], [W11] - = 6, [W11] - = 4, [W11] - = 2, [W11 + W12], 无 }
Wyd	用于 DSP 指令的 Y 数据空间预取目的寄存器 ∈ {W4..W7}

表 20-2： 指令集汇总表

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
1	ADD	ADD Acc	累加器相加	1	1	OA,OB,SA,SB
		ADD f	f = f + WREG	1	1	C,DC,N,OV,Z
		ADD f,WREG	WREG = f + WREG	1	1	C,DC,N,OV,Z
		ADD #lit10,Wn	Wd = lit10 + Wd	1	1	C,DC,N,OV,Z
		ADD Wb,Ws,Wd	Wd = Wb + Ws	1	1	C,DC,N,OV,Z
		ADD Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C,DC,N,OV,Z
		ADD Wso,#Slit4,Acc	16 位符号立即数加到累加器	1	1	OA,OB,SA,SB
2	ADDC	ADDC f	f = f + WREG +(C)	1	1	C,DC,N,OV,Z
		ADDC f,WREG	WREG = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC #lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C,DC,N,OV,Z
		ADDC Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C,DC,N,OV,Z
		ADDC Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C,DC,N,OV,Z
3	AND	AND f	f = f .AND. WREG	1	1	N,Z
		AND f,WREG	WREG = f .AND. WREG	1	1	N,Z
		AND #lit10,Wn	Wd = lit10 .AND. Wd	1	1	N,Z
		AND Wb,Ws,Wd	Wd = Wb .AND. Ws	1	1	N,Z
		AND Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N,Z
4	ASR	ASR f	f = 算术右移 f	1	1	C,N,OV,Z
		ASR f,WREG	WREG = 算术右移 f	1	1	C,N,OV,Z
		ASR Ws,Wd	Wd = 算术右移 Ws	1	1	C,N,OV,Z
		ASR Wb,Wns,Wnd	Wnd = 算术右移 Wb，移位位数为 Wns	1	1	N,Z
		ASR Wb,#lit5,Wnd	Wnd = 算术右移 Wb，移位位数为 lit5	1	1	N,Z
5	BCLR	BCLR f,#bit4	位清零 f	1	1	无
		BCLR Ws,#bit4	位清零 Ws	1	1	无
6	BRA	BRA C,Expr	如果有进位，则转移	1	1 (2)	无
		BRA GE,Expr	如果大于或等于，则转移	1	1 (2)	无
		BRA GEU,Expr	如果无符号大于或等于，则转移	1	1 (2)	无
		BRA GT,Expr	如果大于，则转移	1	1 (2)	无
		BRA GTU,Expr	如果无符号大于，则转移	1	1 (2)	无
		BRA LE,Expr	如果小于或等于，则转移	1	1 (2)	无
		BRA LEU,Expr	如果无符号小于或等于，则转移	1	1 (2)	无
		BRA LT,Expr	如果小于，则转移	1	1 (2)	无
		BRA LTU,Expr	如果无符号小于，则转移	1	1 (2)	无
		BRA N,Expr	如果为负，则转移	1	1 (2)	无
		BRA NC,Expr	如果没有进位，则转移	1	1 (2)	无
		BRA NN,Expr	如果不为负，则转移	1	1 (2)	无
		BRA NOV,Expr	如果不溢出，则转移	1	1 (2)	无
		BRA NZ,Expr	如果非零，则转移	1	1 (2)	无
		BRA OA,Expr	如果累加器 A 溢出，则转移	1	1 (2)	无
		BRA OB,Expr	如果累加器 B 溢出，则转移	1	1 (2)	无
		BRA OV,Expr	如果溢出，则转移	1	1 (2)	无
		BRA SA,Expr	如果累加器 A 饱和，则转移	1	1 (2)	无
		BRA SB,Expr	如果累加器 B 饱和，则转移	1	1 (2)	无
		BRA Expr	无条件转移	1	2	无
		BRA Z,Expr	如果为零，则转移	1	1 (2)	无
		BRA Wn	计算转移	1	2	无
7	BSET	BSET f,#bit4	位置位 f	1	1	无
		BSET Ws,#bit4	位置位 Ws	1	1	无
8	BSW	BSW.C Ws,Wb	把 C 位写入 Ws<Wb>	1	1	无
		BSW.Z Ws,Wb	把 Z 位写入 Ws<Wb>	1	1	无
9	BTG	BTG f,#bit4	位翻转 f	1	1	无
		BTG Ws,#bit4	位翻转 Ws	1	1	无
10	BTSC	BTSC f,#bit4	位测试 f，如果清零则跳过	1	1 (2 或 3)	无
		BTSC Ws,#bit4	位测试 Ws，如果清零则跳过	1	1 (2 或 3)	无

表 **20-2**： 指令集汇总表 （续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
11	BTSS	BTSS f,#bit4	位测试 f，如果为 1 则跳过	1	1 (2 或 3)	无
		BTSS Ws,#bit4	位测试 Ws，如果为 1 则跳过	1	1 (2 或 3)	无
12	BTST	BTST f,#bit4	位测试 f	1	1	Z
		BTST.C Ws,#bit4	位测试 Ws，并将被测试位拷贝至 C	1	1	C
		BTST.Z Ws,#bit4	位测试 Ws，并将被测试位的反码拷贝至 Z	1	1	Z
		BTST.C Ws,Wb	位测试 Ws<Wb>，并将被测试位拷贝至 C	1	1	C
		BTST.Z Ws,Wb	位测试 Ws<Wb>，并将被测试位的反码拷贝至 Z	1	1	Z
13	BTSTS	BTSTS f,#bit4	位测试，然后置 1 f	1	1	Z
		BTSTS.C Ws,#bit4	位测试 Ws，将被测试位拷贝至 C，随后将被测试位置 1	1	1	C
		BTSTS.Z Ws,#bit4	位测试 Ws，将被测试位的反码拷贝至 Z，随后将被测试位置 1	1	1	Z
14	CALL	CALL lit23	调用子程序	2	2	无
		CALL Wn	间接调用子程序	1	2	无
15	CLR	CLR f	f = 0x0000	1	1	无
		CLR WREG	WREG = 0x0000	1	1	无
		CLR Ws	Ws = 0x0000	1	1	无
		CLR Acc,Wx,Wxd,Wy,Wyd,AWB	清零累加器	1	1	OA,OB,SA,SB
16	CLRWDT	CLRWDT	清零看门狗定时器	1	1	WDTO,Sleep
17	COM	COM f	f = $\overline{f}$	1	1	N,Z
		COM f,WREG	WREG = $\overline{f}$	1	1	N,Z
		COM Ws,Wd	Wd = $\overline{Ws}$	1	1	N,Z
18	CP	CP f	比较 f 与 WREG	1	1	C,DC,N,OV,Z
		CP Wb,#lit5	比较 Wb 与 lit5	1	1	C,DC,N,OV,Z
		CP Wb,Ws	比较 Wb 与 Ws （Wb - Ws）	1	1	C,DC,N,OV,Z
19	CP0	CP0 f	比较 f 与 0xFFFF	1	1	C,DC,N,OV,Z
		CP0 Ws	比较 Ws 与 0xFFFF	1	1	C,DC,N,OV,Z
20	CPB	CPB f	比较 f 与 WREG，带借位	1	1	C,DC,N,OV,Z
		CPB Wb,#lit5	比较 Wb 与 lit5，带借位	1	1	C,DC,N,OV,Z
		CPB Wb,Ws	比较 Wb 与 Ws，带借位 （Wb - Ws - $\overline{C}$）	1	1	C,DC,N,OV,Z
21	CPSEQ	CPSEQ Wb, Wn	比较 Wb 与 Wn，如果 相等，则跳过	1	1 (2 或 3)	无
22	CPSGT	CPSGT Wb, Wn	比较 Wb 与 Wn，如果大于，则跳过	1	1 (2 或 3)	无
23	CPSLT	CPSLT Wb, Wn	比较 Wb 与 Wn，如果小于，则跳过	1	1 (2 或 3)	无
24	CPSNE	CPSNE Wb, Wn	比较 Wb 与 Wn，如果不相等，则跳过	1	1 (2 或 3)	无
25	DAW	DAW Wn	Wn = 十进制调整 Wn	1	1	C
26	DEC	DEC f	f = f -1	1	1	C,DC,N,OV,Z
		DEC f,WREG	WREG = f -1	1	1	C,DC,N,OV,Z
		DEC Ws,Wd	Wd = Ws - 1	1	1	C,DC,N,OV,Z
27	DEC2	DEC2 f	f = f -2	1	1	C,DC,N,OV,Z
		DEC2 f,WREG	WREG = f -2	1	1	C,DC,N,OV,Z
		DEC2 Ws,Wd	Wd = Ws - 2	1	1	C,DC,N,OV,Z
28	DISI	DISI #lit14	禁止中断，且持续时间为 lit14 个指令周期	1	1	无
29	DIV	DIV.S Wm,Wn	有符号 16/16 位整数除法	1	18	N,Z,C,OV
		DIV.SD Wm,Wn	有符号 32/16 位整数除法	1	18	N,Z,C,OV
		DIV.U Wm,Wn	无符号 16/16 位整数除法	1	18	N,Z,C,OV
		DIV.UD Wm,Wn	无符号 32/16 位整数除法	1	18	N,Z,C,OV
30	DIVF	DIVF Wm,Wn	有符号 16/16 位小数除法	1	18	N,Z,C,OV
31	DO	DO #lit14,Expr	执行 DO 循环到 PC+Expr，执行次数为 lit14+1 次	2	2	无
		DO Wn,Expr	执行 DO 循环到 PC+Expr，执行次数为 (Wn) + 1 次	2	2	无
32	ED	ED Wm*Wm,Acc,Wx,Wy,Wxd	欧几里德距离 （无累加）	1	1	OA,OB,OAB,SA,SB,SAB
33	EDAC	EDAC Wm*Wm,Acc,Wx,Wy,Wxd	欧几里德距离	1	1	OA,OB,OAB,SA,SB,SAB
34	EXCH	EXCH Wns,Wnd	交换 Wns 与 Wnd 的内容	1	1	无

表 20-2： 指令集汇总表（续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
35	FBCL	FBCL Ws,Wnd	搜索自左起第一个位变化	1	1	C
36	FF1L	FF1L Ws,Wnd	搜索自左起第一个 1	1	1	C
37	FF1R	FF1R Ws,Wnd	搜索自右起第一个 1	1	1	C
38	GOTO	GOTO Expr	转移至地址	2	2	无
		GOTO Wn	间接转移	1	2	无
39	INC	INC f	f = f + 1	1	1	C,DC,N,OV,Z
		INC f,WREG	WREG = f + 1	1	1	C,DC,N,OV,Z
		INC Ws,Wd	Wd = Ws + 1	1	1	C,DC,N,OV,Z
40	INC2	INC2 f	f = f + 2	1	1	C,DC,N,OV,Z
		INC2 f,WREG	WREG = f + 2	1	1	C,DC,N,OV,Z
		INC2 Ws,Wd	Wd = Ws + 2	1	1	C,DC,N,OV,Z
41	IOR	IOR f	f = f .IOR. WREG	1	1	N,Z
		IOR f,WREG	WREG = f .IOR. WREG	1	1	N,Z
		IOR #lit10,Wn	Wd = lit10 .IOR. Wd	1	1	N,Z
		IOR Wb,Ws,Wd	Wd = Wb .IOR. Ws	1	1	N,Z
		IOR Wb,#lit5,Wd	Wd = Wb .IOR. lit5	1	1	N,Z
42	LAC	LAC Wso,#Slit4,Acc	装载累加器	1	1	OA,OB,OAB, SA,SB,SAB
43	LNK	LNK #lit14	分配堆栈帧	1	1	无
44	LSR	LSR f	f = 逻辑右移 f	1	1	C,N,OV,Z
		LSR f,WREG	WREG = 逻辑右移 f	1	1	C,N,OV,Z
		LSR Ws,Wd	Wd = 逻辑右移 Ws	1	1	C,N,OV,Z
		LSR Wb,Wns,Wnd	Wnd = 逻辑右移 Wb，移位位数 Wns	1	1	N,Z
		LSR Wb,#lit5,Wnd	Wnd = 逻辑右移 Wb，移位位数 lit5	1	1	N,Z
45	MAC	MAC Wm*Wn,Acc,Wx,Wxd,Wy,Wyd, AWB	相乘并累加	1	1	OA,OB,OAB, SA,SB,SAB
		MAC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	平方并累加	1	1	OA,OB,OAB, SA,SB,SAB
46	MOV	MOV f,Wn	传送 f 至 Wn	1	1	无
		MOV f	传送 f 至 f	1	1	N,Z
		MOV f,WREG	传送 f 至 WREG	1	1	N,Z
		MOV #lit16,Wn	传送 16 位立即数至 Wn	1	1	无
		MOV.b #lit8,Wn	传送 8 位立即数至 Wn	1	1	无
		MOV Wn,f	传送 Wn 至 f	1	1	无
		MOV Wso,Wdo	传送 Ws 至 Wd	1	1	无
		MOV WREG,f	传送 WREG 至 f	1	1	N,Z
		MOV.D Wns,Wd	双字传送 W(ns):W(ns + 1) 至 Wd	1	2	无
		MOV.D Ws,Wnd	双字传送 Ws 至 W(nd + 1):W(nd)	1	2	无
47	MOVSAC	MOVSAC Acc,Wx,Wxd,Wy,Wyd,AWB	预取并保存累加器	1	1	无
48	MPY	MPY Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	Wm 与 Wn 相乘，结果存入累加器	1	1	OA,OB,OAB, SA,SB,SAB
		MPY Wm*Wm,Acc,Wx,Wxd,Wy,Wyd	Wm 平方，结果存入累加器	1	1	OA,OB,OAB, SA,SB,SAB
49	MPY.N	MPY.N Wm*Wn,Acc,Wx,Wxd,Wy,Wyd	-(Wm × Wn)，结果存入累加器	1	1	无
50	MSC	MSC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd, AWB	相乘并从累加器中减去	1	1	OA,OB,OAB, SA,SB,SAB
51	MUL	MUL.SS Wb,Ws,Wnd	{Wnd + 1, Wnd} = signed(Wb) * signed(Ws)	1	1	无
		MUL.SU Wb,Ws,Wnd	{Wnd + 1, Wnd} = signed(Wb) * unsigned(Ws)	1	1	无
		MUL.US Wb,Ws,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * signed(Ws)	1	1	无
		MUL.UU Wb,Ws,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * unsigned(Ws)	1	1	无
		MUL.SU Wb,#lit5,Wnd	{Wnd + 1, Wnd} = signed(Wb) * unsigned(lit5)	1	1	无
		MUL.UU Wb,#lit5,Wnd	{Wnd + 1, Wnd} = unsigned(Wb) * unsigned(lit5)	1	1	无
		MUL f	W3:W2 = f * WREG	1	1	无

表 **20-2**： 指令集汇总表 （续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
52	NEG	NEG Acc	累加器求补	1	1	OA,OB,OAB, SA,SB,SAB
		NEG f	f = $\overline{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG f,WREG	WREG = $\overline{f}$ + 1	1	1	C,DC,N,OV,Z
		NEG Ws,Wd	Wd = $\overline{Ws}$ + 1	1	1	C,DC,N,OV,Z
53	NOP	NOP	空操作	1	1	无
		NOPR	空操作	1	1	无
54	POP	POP f	从栈顶 （TOS）弹出至 f	1	1	无
		POP Wdo	从栈顶 （TOS）弹出至 Wdo	1	1	无
		POP.D Wnd	从栈顶 （TOS）弹出至 W(nd):W(nd+1)	1	2	无
		POP.S	把影子寄存器弹出堆栈	1	1	全部
55	PUSH	PUSH f	把 f 压入栈顶 （TOS）	1	1	无
		PUSH Wso	把 Wso 压入栈顶 （TOS）	1	1	无
		PUSH.D Wns	把 W(ns):W(ns+1) 压入栈顶 （TOS）	1	2	无
		PUSH.S	把影子寄存器压入堆栈	1	1	无
56	PWRSAV	PWRSAV #lit1	进入休眠或空闲模式	1	1	WDTO,Sleep
57	RCALL	RCALL Expr	相对调用	1	2	无
		RCALL Wn	计算调用	1	2	无
58	REPEAT	REPEAT #lit14	重复执行下一条指令 lit14 + 1 次	1	1	无
		REPEAT Wn	重复执行下一条指令 (Wn) + 1 次	1	1	无
59	RESET	RESET	软件器件复位	1	1	无
60	RETFIE	RETFIE	从中断返回	1	3 (2)	无
61	RETLW	RETLW #lit10,Wn	返回，并将立即数存储至 Wn	1	3 (2)	无
62	RETURN	RETURN	从子程序返回	1	3 (2)	无
63	RLC	RLC f	f = 带进位位循环左移 f	1	1	C,N,Z
		RLC f,WREG	WREG = 带进位位循环左移 f	1	1	C,N,Z
		RLC Ws,Wd	Wd = 带进位位循环左移 Ws	1	1	C,N,Z
64	RLNC	RLNC f	f = 不带进位位循环左移 f	1	1	N,Z
		RLNC f,WREG	WREG = 不带进位位循环左移 f	1	1	N,Z
		RLNC Ws,Wd	Wd = 不带进位位循环左移 Ws	1	1	N,Z
65	RRC	RRC f	f = 带进位位循环右移 f	1	1	C,N,Z
		RRC f,WREG	WREG = 带进位位循环右移 f	1	1	C,N,Z
		RRC Ws,Wd	Wd = 带进位位循环右移 Ws	1	1	C,N,Z
66	RRNC	RRNC f	f = 不带进位位循环右移 f	1	1	N,Z
		RRNC f,WREG	WREG = 不带进位位循环右移 f	1	1	N,Z
		RRNC Ws,Wd	Wd = 不带进位位循环右移 Ws	1	1	N,Z
67	SAC	SAC Acc,#Slit4,Wdo	保存累加器	1	1	无
		SAC.R Acc,#Slit4,Wdo	保存舍入后的累加器	1	1	无
68	SE	SE Ws,Wnd	Wnd = 符号扩展后的 Ws	1	1	C,N,Z
69	SETM	SETM f	f = 0xFFFF	1	1	无
		SETM WREG	WREG = 0xFFFF	1	1	无
		SETM Ws	Ws = 0xFFFF	1	1	无
70	SFTAC	SFTAC Acc,Wn	算术移位累加器 (Wn) 次	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC Acc,#Slit6	算术移位累加器 Slit6 次	1	1	OA,OB,OAB, SA,SB,SAB
71	SL	SL f	f = 左移 f	1	1	C,N,OV,Z
		SL f,WREG	WREG = 左移 f	1	1	C,N,OV,Z
		SL Ws,Wd	Wd = 左移 Ws	1	1	C,N,OV,Z
		SL Wb,Wns,Wnd	Wnd = 左移 Wb，且移位位数为 Wns	1	1	N,Z
		SL Wb,#lit5,Wnd	Wnd = 左移 Wb，且移位位数为 lit5	1	1	N,Z

表 20-2：　指令集汇总表 （续）

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
72	SUB	SUB Acc	从累加器减去	1	1	OA,OB,OAB, SA,SB,SAB
		SUB f	f = f - WREG	1	1	C,DC,N,OV,Z
		SUB f,WREG	WREG = f - WREG	1	1	C,DC,N,OV,Z
		SUB #lit10,Wn	Wn = Wn - lit10	1	1	C,DC,N,OV,Z
		SUB Wb,Ws,Wd	Wd = Wb - Ws	1	1	C,DC,N,OV,Z
		SUB Wb,#lit5,Wd	Wd = Wb - lit5	1	1	C,DC,N,OV,Z
73	SUBB	SUBB f	f = f - WREG - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB f,WREG	WREG = f - WREG - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB #lit10,Wn	Wn = Wn - lit10 - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB Wb,Ws,Wd	Wd = Wb - Ws - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBB Wb,#lit5,Wd	Wd = Wb - lit5 - ($\overline{C}$)	1	1	C,DC,N,OV,Z
74	SUBR	SUBR f	f = WREG - f	1	1	C,DC,N,OV,Z
		SUBR f,WREG	WREG = WREG - f	1	1	C,DC,N,OV,Z
		SUBR Wb,Ws,Wd	Wd = Ws - Wb	1	1	C,DC,N,OV,Z
		SUBR Wb,#lit5,Wd	Wd = lit5 - Wb	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR f	f = WREG - f - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR f,WREG	WREG = WREG -f - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR Wb,Ws,Wd	Wd = Ws - Wb - ($\overline{C}$)	1	1	C,DC,N,OV,Z
		SUBBR Wb,#lit5,Wd	Wd = lit5 - Wb - ($\overline{C}$)	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b Wn	Wn = 半字节交换 Wn	1	1	无
		SWAP Wn	Wn = 字节交换 Wn	1	1	无
77	TBLRDH	TBLRDH Ws,Wd	读程序存储单元 <23:16> 至 Wd<7:0>	1	2	无
78	TBLRDL	TBLRDL Ws,Wd	读程序存储单元 <15:0> 至 Wd	1	2	无
79	TBLWTH	TBLWTH Ws,Wd	写 Ws<7:0> 至程序存储单元 <23:16>	1	2	无
80	TBLWTL	TBLWTL Ws,Wd	写 Ws 至程序存储单元 <15:0>	1	2	无
81	ULNK	ULNK	释放堆栈帧	1	1	无
82	XOR	XOR f	f = f .XOR. WREG	1	1	N,Z
		XOR f,WREG	WREG = f .XOR. WREG	1	1	N,Z
		XOR #lit10,Wn	Wd = lit10 .XOR. Wd	1	1	N,Z
		XOR Wb,Ws,Wd	Wd = Wb .XOR. Ws	1	1	N,Z
		XOR Wb,#lit5,Wd	Wd = Wb .XOR. lit5	1	1	N,Z
83	ZE	ZE Ws,Wnd	Wnd = 零扩展后的 Ws	1	1	C,Z,N

21.0 开发支持

一系列硬件及软件开发工具对 PIC® 单片机提供支持：

- 集成开发环境
  - MPLAB® IDE 软件
- 汇编器 / 编译器 / 链接器
  - MPASM™ 汇编器
  - MPLAB C18 和 MPLAB C30 C 编译器
  - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
  - MPLAB ASM30 汇编器 / 链接器 / 库
- 模拟器
  - MPLAB SIM 软件模拟器
- 仿真器
  - MPLAB ICE 2000 在线仿真器
  - MPLAB REAL ICE™ 在线仿真器
- 在线调试器
  - MPLAB ICD 2
- 器件编程器
  - PICSTART® Plus 开发编程器
  - MPLAB PM3 器件编程器
  - PICkit™ 2 开发编程器
- 低成本演示和开发板及评估工具包

21.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16 位单片机市场提供了前所未有的易于使用的软件开发平台。MPLAB IDE 是基于 Windows® 操作系统的应用软件，包括：

- 一个包含所有调试工具的图形界面
  - 模拟器
  - 编程器 （单独销售）
  - 仿真器 （单独销售）
  - 在线调试器 （单独销售）
- 具有彩色上下文代码显示的全功能编辑器
- 多项目管理器
- 内容可直接编辑的可定制式数据窗口
- 高级源代码调试
- 可视化器件初始化程序，便于进行寄存器的初始化
- 鼠标停留在变量上进行查看的功能
- 通过拖放把变量从源代码窗口拉到观察窗口
- 丰富的在线帮助
- 集成了可选的第三方工具，如 HI-TECH 软件 C 编译器和 IAR C 编译器

MPLAB IDE 可以让您：

- 编辑源文件 （汇编语言或 C 语言）
- 点击一次即可完成汇编 （或编译）并将代码下载到 PIC MCU 仿真器和模拟器工具中 （自动更新所有项目信息）
- 可使用如下各项进行调试：
  - 源文件 （汇编语言或 C 语言）
  - 混合汇编语言和 C 语言
  - 机器码

MPLAB IDE 在单个开发范例中支持使用多种调试工具，包括从成本效益高的模拟器到低成本的在线调试器，再到全功能的仿真器。这样缩短了用户升级到更加灵活而功能更强大的工具时的学习时间。

21.2 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于所有的 PIC MCU。

MPASM 汇编器可生成用于 MPLINK 目标链接器的可重定位目标文件、Intel® 标准 HEX 文件、详细描述存储器使用状况和符号参考的 MAP 文件、包含源代码行及生成机器码的绝对 LST 文件以及用于调试的 COFF 文件。

MPASM 汇编器具有如下特征：

- 集成在 MPLAB IDE 项目中
- 用户定义的宏可简化汇编代码
- 对多用途源文件进行条件汇编
- 允许完全控制汇编过程的指令

21.3 MPLAB C18 和 MPLAB C30 C 编译器

MPLAB C18 和 MPLAB C30 代码开发系统是完全的 ANSI C 编译器，分别适用于 Microchip 的 PIC18 和 PIC24 系列单片机及 dsPIC30F 和 dsPIC33 系列数字信号控制器。这些编译器可提供其他编译器并不具备的强大的集成功能和出众的代码优化能力，且使用方便。

为便于源代码调试，编译器提供了针对 MPLAB IDE 调试器的优化符号信息。

21.4 MPLINK 目标链接器 / MPLIB 目标库管理器

MPLINK 目标链接器包含了由 MPASM 汇编器、MPLAB C18 C 编译器产生的可重定位目标。通过使用链接器脚本中的指令，它还可链接预编译库中的可重定位目标。

MPLIB目标库管理器管理预编译代码库文件的创建和修改。当从源文件调用库中的一段子程序时，只有包含此子程序的模块被链接到应用中。这样可使大型库在许多不同应用中被高效地利用。

目标链接器 / 库管理器具有如下特征：

- 高效地连接单个的库而不是许多小文件
- 通过将相关的模块组合在一起来增强代码的可维护性
- 只要列出、替换、删除和抽取模块，便可灵活地创建库

21.5 MPLAB ASM30 汇编器、链接器和库管理器

MPLAB ASM30 汇编器为 dsPIC30F 器件提供转换自符号汇编语言的可重定位机器码。 MPLAB C30 C 编译器使用该汇编器生成目标文件。汇编器产生可重定位目标文件之后，可将这些目标文件存档，或与其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特征：

- 支持整个 dsPIC30F 指令集
- 支持定点数据和浮点数据
- 命令行界面
- 丰富的指令集
- 灵活的宏语言
- MPLAB IDE 兼容性

21.6 MPLAB SIM 软件模拟器

MPLAB SIM软件模拟器在指令级对PIC MCU和dsPIC® DSC 进行模拟，使得用户可以在 PC 主机的环境下进行代码开发。对于任何给定的指令，用户均可对数据区进行检查或修改，并通过各种触发机制来产生激励。可以将各寄存器的情况记录在文件中，以便进行进一步地运行时分析。跟踪缓冲器和逻辑分析器的显示使模拟器还能记录和跟踪程序的执行、 I/O 的动作、大部分的外设及内部寄存器的状况。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C18 和 MPLAB C30 C 编译器以及MPASM和MAPLAB ASM30 汇编器的符号调试。该软件模拟器可用于在硬件实验室环境外灵活地开发和调试代码，是一款完美且经济的软件开发工具。

21.7 MPLAB ICE 2000 高性能在线仿真器

MPLAB ICE 2000 在线仿真器旨在为产品开发工程师提供一整套用于 PIC 单片机的设计工具。 MPLAB ICE 2000在线仿真器的软件控制由MPLAB集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 2000 是全功能仿真器系统，它具有增强的跟踪、触发和数据监控功能。处理器模块可插拔，使系统可轻松进行重新配置以适应各种不同处理器的仿真需要。MPLAB ICE 2000 在线仿真器的架构允许对其进行扩展以支持新的 PIC 单片机。

MPLAB ICE 2000 在线仿真器系统设计为一款实时仿真系统，该仿真系统具备通常只有昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft® Windows® 32 位操作系统可使这些功能在一个简单而统一的应用中得到很好的利用。

21.8 MPLAB REAL ICE 在线仿真器系统

MPLAB REAL ICE在线仿真器系统是Microchip针对其闪存 DSC 和 MCU 器件而推出的新一代高速仿真器。结合 MPLAB 集成开发环境（IDE）所具有的易于使用且功能强大的图形用户界面，该仿真器可对 PIC® 闪存 MCU 和 dsPIC® DSC进行调试和编程。IDE是随每个工具包一起提供的。

MPLAB REAL ICE 探针通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与常用 MPLAB ICD 2 系统兼容的连接器 （RJ11）或新型抗噪声、高速低压差分信号（LVDS）互连电缆 （CAT5）与目标板相连。

可通过 MPLAB IDE 下载将来版本的固件，对 MPLAB REAL ICE 进行现场升级。在即将推出的 MPLAB IDE 版本中，会支持许多新器件，还将增加一些新特性，如软件断点和汇编代码跟踪等。在同类仿真器中， MPLAB REAL ICE 的优势十分明显：低成本、高速仿真、实时变量监视、跟踪分析、复杂断点、耐用的探针接口及较长（长达 3 米）的互连电缆。

21.9 MPLAB ICD 2 在线调试器

Microchip 的在线调试器 MPLAB ICD 2 是一款功能强大而成本低廉的运行时开发工具，通过 RS-232 或高速 USB 接口与 PC 主机相连。该工具基于闪存 PIC MCU，可用于开发本系列及其他 PIC MCU 和 dsPIC DSC。MPLAB ICD 2使用了闪存器件中内建的在线调试功能。该功能结合Microchip的在线串行编程（In-Circuit Serial Programming™， ICSP™）协议，可在 MPLAB 集成开发环境的图形用户界面上提供成本效益很高的在线闪存调试。这使设计人员可通过设置断点、单步运行以及对变量、CPU 状态以及外设寄存器进行监视的方法实现源代码的开发和调试。其全速运行特性可对硬件和应用进行实时测试。 MPLAB ICD 2 还可用作某些 PIC 器件的开发编程器。

21.10 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款通用的、符合 CE 规范的器件编程器，其可编程电压设置在 VDDMIN 和 VDDMAX 之间时可靠性最高。它有一个用来显示菜单和错误信息的大 LCD 显示器（128 x 64），以及一个支持各种封装类型的可拆卸模块化插槽装置。编程器标准配置中带有一根 ICSP™ 电缆。在单机模式下， MPLAB PM3 器件编程器不必与 PC 相连即可对 PIC 器件进行读取、验证和编程。在该模式下它还可设置代码保护。 MPLAB PM3 通过 RS-232 或 USB 电缆连接到 PC 主机上。MPLAB PM3 具备高速通信能力以及优化算法，可对存储器很大的器件进行快速编程，它还采用 SD/MMC 卡用作文件存储及数据安全应用。

21.11 PICSTART Plus 开发编程器

PICSTART Plus开发编程器是一款易于使用而成本低廉的原型编程器。它通过 COM （RS-232）端口与 PC 相连。 MPLAB 集成开发环境软件使得该编程器的使用简便、高效。PICSTART Plus 开发编程器支持采用 DIP 封装的大部分 PIC 器件，其引脚数最多可达 40 个。引脚数更多的器件，如 PIC16C92X 和 PIC17C76X，可通过连接一个转接插槽来获得支持。PICSTART Plus 开发编程器符合 CE 规范。

21.12 PICkit 2 开发编程器

PICkit™ 2 开发编程器是一个低成本编程器；对于某些选定闪存器件，它也是一个调试器，通过其易于使用的接口可对众多 Microchip 的低档、中档和 PIC18F 系列闪存单片机进行编程。 PICkit 2 入门工具包中包含一个有实验布线区的开发板、十二堂系列课程、软件和 HI-TECH 的 PICC™ Lite C 编译器，有助于用户快速掌握 PIC® 单片机的使用。这一工具包为使用 Microchip 功能强大的中档闪存系列单片机进行编程、评估和应用开发，提供了所需的一切。

21.13 演示、开发和评估板

有许多演示、开发和评估板可用于各种 PIC MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于测试和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、 RS-232 接口、 LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外， Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、 KEELOQ® 数据安全产品 IC、CAN、IrDA®、PowerSmart 电池管理、SEEVAL® 评估系统、 Σ–Δ ADC、流速传感器，等等。

有关演示、开发和评估工具包的完整列表，请查阅 Microchip 公司网页 （www.microchip.com）。

22.0 电气特性

本章将对 dsPIC30F 电气特性进行概括介绍。其余信息将在该文档的后续版本中给出。

有关 dsPIC30F 架构和内核的详细信息，请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）。

下面列出了 dsPIC30F 系列器件的绝对最大额定值。器件长时间工作在最大额定值条件可能影响其可靠性。我们不建议使器件在或超过本规范指定的最大额定值条件下运行。

绝对最大额定值 （腷©

环境温度 ....-40°C 至 +125°C
存储温度 .... -65°C 至 +150°C
任一引脚相对于 Vss 的电压 （除 VDD 和 $\overline{\text{MCLR}}$ 外）........................ -0.3V 至 （VDD + 0.3V）
VDD 引脚相对于 Vss 的电压 .... -0.3V 至 +5.5V
MCLR 引脚相对于 Vss 的电压 （注 **1**）...................................... 0V 至 +13.25V
总功耗 （注 **2**）.. 1.0W
Vss 引脚的最大电流 .... 300 mA
VDD 引脚的最大电流 .... 250 mA
输入箝位电流， IIK （VI < 0 或 VI > VDD）..................................... ±20 mA
输出箝位电流， IOK （VO < 0 或 VO > VDD）.................................... ±20 mA
任一 I/O 引脚最大输出灌电流 .... 25 mA
任一 I/O 引脚最大输出拉电流 .... 25 mA
全部端口总的灌电流 .... 200 mA
全部端口总的拉电流 .... 200 mA

注 **1**：MCLR/VPP 引脚上的尖峰电压低于 Vss，导致感应电流超过 80 mA 时，可能引起锁死。因此，当把一个“低”电平加到 MCLR/VPP 引脚上时，应串联一个 50-100Ω 的电阻，而不要将这个引脚直接接到 VDD。

2：最大可允许电流由器件最大功耗决定，参见表 22-2。

† 注：如果器件工作条件超过上述 “绝对最大额定值”，可能引起器件永久性损坏。这仅是极限参数，我们不建议器件工作在极限值甚至超过上述极限值。器件长时间工作在极限条件下可能会影响其可靠性。

22.1 DC 特性

表 22-1： 工作 MIPS 与电压

VDD 范围	温度范围	最大 MIPS	
		dsPIC30F2010-30I	dsPIC30F2010-20E
4.5-5.5V	-40°C 至 85°C	30	—
4.5-5.5V	-40°C 至 125°C	—	20
3.0-3.6V	-40°C 至 85°C	20	—
3.0-3.6V	-40°C 至 125°C	—	15
2.5-3.0V	-40°C 至 85°C	10	—

表 22-2： 热工作条件

额定值	符号	最小值	典型值	最大值	单位
dsPIC30F2010-30I					
工作结温范围	T_J	-40		+125	°C
工作环境温度范围	T_A	-40		+85	°C
dsPIC30F2010-20E					
工作结温范围	T_J	-40		+150	°C
工作环境温度范围	T_A	-40		+125	°C
功耗： 内部芯片功耗： $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O 引脚功耗： $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	P_D	$P_{INT} + P_{I/O}$			W
最大允许功耗	P_{DMAX}	$(T_J - T_A) / \theta_{JA}$			W

表 22-3： 热封装特性

特性	符号	典型值	最大值	单位	注
封装热阻， 28 引脚 SOIC （SO）	θ_{JA}	48.3		°C/W	1
封装热阻， 28 引脚 QFN	θ_{JA}	33.7		°C/W	1
封装热阻， 28 引脚 SPDIP （SP）	θ_{JA}	42		°C/W	1

注 1： 通过封装模拟获得结点与环境的热阻值 θ_{JA}。

表 22-4： DC 温度和电压规范

DC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度 工业级：$-40°C \le T_A \le +85°C$ 扩展级：$-40°C \le T_A \le +125°C$				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
工作温度 (2)							
DC10	V_{DD}	电源电压	2.5	—	5.5	V	工业级温度
DC11	V_{DD}	电源电压	3.0	—	5.5	V	扩展级温度
DC12	V_{DR}	RAM 数据保持电压 (3)	—	1.5	—	V	
DC16	V_{POR}	确保内部上电复位信号的 V_{DD} 起始电压	—	V_{SS}	—	V	
DC17	S_{VDD}	确保内部上电复位信号的 V_{DD} 上升率	0.05			V/ms	0-5V/0.1 s 0-3V/60 ms

注 1： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。
2： 这些参数为特性值，但生产时未经测试。
3： 这是在不丢失 RAM 数据的前提下， V_{DD} 的下限值。

表 22-5： DC 特性：工作电流 （IDD）

DC 特性			标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C			
参数编号	典型值	最大值	单位	条件		
工作电流 （IDD） [1]						
DC31a	1.6	3	mA	25°C	3.3V	0.128 MIPS LPRC （512 kHz）
DC31b	1.6	3	mA	85°C		
DC31c	1.6	3	mA	125°C		
DC31e	3.9	7	mA	25°C	5V	
DC31f	3.5	7	mA	85°C		
DC31g	3.4	7	mA	125°C		
DC30a	3	5	mA	25°C	3.3V	（1.8 MIPS） FRC （7.37MHz）
DC30b	3	5	mA	85°C		
DC30c	3	5	mA	125°C		
DC30e	6	9	mA	25°C	5V	
DC30f	6	9	mA	85°C		
DC30g	6	9	mA	125°C		
DC23a	9	14	mA	25°C	3.3V	4 MIPS EC 模式， 4X PLL
DC23b	10	15	mA	85°C		
DC23c	10	15	mA	125°C		
DC23e	16	24	mA	25°C	5V	
DC23f	16	24	mA	85°C		
DC23g	16	24	mA	125°C		
DC24a	21	32	mA	25°C	3.3V	10 MIPS EC 模式， 4X PLL
DC24b	21	32	mA	85°C		
DC24c	21	32	mA	125°C		
DC24e	35	53	mA	25°C	5V	
DC24f	36	53	mA	85°C		
DC24g	36	53	mA	125°C		
DC27a	39	59	mA	25°C	3.3V	20 MIPS EC 模式， 8X PLL
DC27b	39	59	mA	85°C		
DC27d	66	99	mA	25°C	5V	
DC27e	66	99	mA	85°C		
DC27f	66	99	mA	125°C		
DC29a	95	150	mA	25°C	5V	30 MIPS EC 模式， 16X PLL
DC29b	94	150	mA	85°C		

注 1： 供电电流主要受工作电压和频率的影响。其他因素如 I/O 引脚负载和开关频率、振荡器类型、内部代码执行模式和温度也对电流消耗有影响。所有 IDD 测量的测试条件为：OSC1 使用满幅的外部方波进行驱动。所有 I/O 引脚配置为输入且被拉到 VDD。MCLR = VDD， WDT、 FSCM、 LVD 和 BOR 被禁止。 CPU、SRAM、程序存储器和数据存储器处于工作状态。外设模块均不工作。

表 22-6： DC 特性：空闲电流 （IIDLE）

DC 特性			标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C			
参数编号	典型值 (1)	最大值	单位	条件		
空闲电流 （IIDLE） (2)						
DC51a	1.5	3.0	mA	25°C	3.3V	0.128 MIPS LPRC （512 kHz）
DC51b	1.5	3.0	mA	85°C		
DC51c	1.5	3.0	mA	125°C		
DC51e	4.1	7	mA	25°C	5V	
DC51f	3.6	7	mA	85°C		
DC51g	3.5	7	mA	125°C		
DC50a	3	5	mA	25°C	3.3V	（1.8 MIPS） FRC （7.37MHz）
DC50b	3	5	mA	85°C		
DC50c	3	5	mA	125°C		
DC50e	7	9	mA	25°C	5V	
DC50f	6	9	mA	85°C		
DC50g	6	9	mA	125°C		
DC43a	5	9	mA	25°C	3.3V	4 MIPS EC 模式， 4X PLL
DC43b	6	9	mA	85°C		
DC43c	6	9	mA	125°C		
DC43e	10	15	mA	25°C	5V	
DC43f	10	15	mA	85°C		
DC43g	10	15	mA	125°C		
DC44a	11	18	mA	25°C	3.3V	10 MIPS EC 模式， 4X PLL
DC44b	12	18	mA	85°C		
DC44c	12	18	mA	125°C		
DC44e	20	30	mA	25°C	5V	
DC44f	20	30	mA	85°C		
DC44g	20	30	mA	125°C		
DC47a	16.8	—	mA	25°C	3.3V	20 MIPS EC 模式， 8X PLL
DC47b	—	—	mA	85°C		
DC47d	30.5	—	mA	25°C	5V	
DC47e	—	—	mA	85°C		
DC47f	—	—	mA	125°C		
DC49a	49	65	mA	25°C	5V	30 MIPS EC 模式， 16X PLL
DC49b	50	65	mA	85°C		

注 1： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。

2： 基本 IIDLE 电流测量是在内核不工作、时钟工作而所有外设模块关闭的条件下进行的。

表 **22-7**： **DC** 特性：掉电电流 （**IPD**）

DC 特性			标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度 工业级：-40°C ≤ T_A ≤ +85°C 扩展级：-40°C ≤ T_A ≤ +125°C			
参数编号	典型值	最大值	单位	条件		
掉电电流 （**IPD**）[1]						
DC60a	0.05	—	μA	25°C	3.3V	基本掉电电流 [2]
DC60b	3	25	μA	85°C		
DC60c	20	50	μA	125°C		
DC60e	0.1	—	μA	25°C	5V	
DC60f	6	35	μA	85°C		
DC60g	40	70	μA	125°C		
DC61a	30	45	μA	25°C	3.3V	看门狗定时器电流：ΔIWDT[2]
DC61b	34	51	μA	85°C		
DC61c	46	69	μA	125°C		
DC61e	35	53	μA	25°C	5V	
DC61f	39	59	μA	85°C		
DC61g	40	60	μA	125°C		
DC62a	4	10	μA	25°C	3.3V	Timer 1，带 32 kHz 晶振：ΔITI32[2]
DC62b	5	10	μA	85°C		
DC62c	4	10	μA	125°C		
DC62e	4	15	μA	25°C	5V	
DC62f	6	15	μA	85°C		
DC62g	5	15	μA	125°C		
DC63a	4	6	μA	25°C	3.3V	BOR 工作：ΔIBOR[2]
DC63b	4	6	μA	85°C		
DC63c	5	7.5	μA	125°C		
DC63e	10	15	μA	25°C	5V	

注 **1**： 基本 IPD 是在所有外设和时钟都关断的条件下进行测量的。所有 I/O 都配置为输入且被拉至高电平。 LVD、BOR 和 WDT 等都关闭。

2： Δ 电流为当模块使能时额外消耗的电流。此电流应被加到基本 IPD 电流。

表 22-8： DC 特性：I/O 引脚输入规范

DC 特性			标准工作条件：2.5V 至 5.5V （（除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
	VIL	输入低电平电压 (2)					
DI10		I/O 引脚： 带施密特触发缓冲器	VSS	—	0.2 VDD	V	
DI15		$\overline{\text{MCLR}}$	VSS	—	0.2 VDD	V	
DI16		OSC1 （处于 XT、 HS 和 LP 模式）	VSS	—	0.2 VDD	V	
DI17		OSC1 （处于 RC 模式） (3)	VSS	—	0.3 VDD	V	
DI18		SDA 和 SCL	VSS	—	0.3 VDD	V	禁止 SM 总线
DI19		SDA 和 SCL	VSS	—	0.2 VDD	V	禁止 SM 总线
	VIH	输入高电平电压 (2)					
DI20		I/O 引脚： 带施密特触发缓冲器	0.8 VDD	—	VDD	V	
DI25		$\overline{\text{MCLR}}$	0.8 VDD	—	VDD	V	
DI26		OSC1 （处于 XT、 HS 和 LP 模式）	0.7 VDD	—	VDD	V	
DI27		OSC1 （处于 RC 模式） (3)	0.9 VDD	—	VDD	V	
DI28		SDA 和 SCL	0.7 VDD	—	VDD	V	禁止 SM 总线
DI29		SDA 和 SCL	0.8 VDD	—	VDD	V	禁止 SM 总线
	ICNPU	**CNxx 上拉电流** (2)					
DI30			50	250	400	μA	VDD = 5V， VPIN = VSS
	IIL	输入泄露电流 (2)(4)(5)					
DI50		I/O 端口	—	0.01	±1	μA	VSS ≤ VPIN ≤ VDD， 引脚处于高阻态
DI51		模拟输入引脚	—	0.50	±1.3	μA	VSS ≤ VPIN ≤ VDD， 引脚处于高阻态
DI55		$\overline{\text{MCLR}}$	—	0.05	±5	μA	VSS ≤ VPIN ≤ VDD
DI56		OSC1	—	0.05	±7	μA	VSS ≤ VPIN ≤ VDD， XT、HS 和 LP 振荡模式

注 1： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。

2： 这些参数为特性值，但生产时未经测试。

3： 在 RC 振荡器配置中， OSC1/CLKI 引脚为施密特触发器输入。不建议在 RC 模式下使用外部时钟驱动 dsPIC30F 器件。

4： $\overline{\text{MCLR}}$ 引脚上的泄露电流主要取决于所加电平。表中给定的电平表示正常工作条件。在不同输入电压条件下可能测得更高的泄露电流。

5： 负电流定义为引脚的拉电流。

表 22-9： DC 特性：I/O 引脚输出规范

DC 特性			标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
	VOL	输出低电平电压 (2)					
DO10		I/O 端口	—	—	0.6	V	IOL = 8.5 mA， VDD = 5V
			—	—	TBD	V	IOL = 2.0 mA， VDD = 3V
DO16		OSC2/CLKO	—	—	0.6	V	IOL = 1.6 mA， VDD = 5V
		（RC 或 EC 振荡模式）	—	—	TBD	V	IOL = 2.0 mA， VDD = 3V
	VOH	输出高电平电压 (2)					
DO20		I/O 端口	VDD – 0.7	—	—	V	IOH = -3.0 mA， VDD = 5V
			TBD	—	—	V	IOH = -2.0 mA， VDD = 3V
DO26		OSC2/CLKO	VDD – 0.7	—	—	V	IOH = -1.3 mA， VDD = 5V
		（RC 或 EC 振荡模式）	TBD	—	—	V	IOH = -2.0 mA， VDD = 3V
		输出引脚的容性负载规范 (2)					
DO50	COSC2	OSC2/SOSC2 引脚	—	—	15	pF	在 XTL、 XT、 HS 和 LP 模式下，使用外部时钟驱动 OSC1 时。
DO56	CIO	所有 I/O 引脚和 OSC2	—	—	50	pF	RC 或 EC 振荡模式
DO58	CB	SCL 和 SDA	—	—	400	pF	在 I^2C 模式下

注 1：除非另外声明，否则 “典型值” 一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。

2：这些参数为特性值，但生产时未经测试。

图 22-1： 欠压复位特性

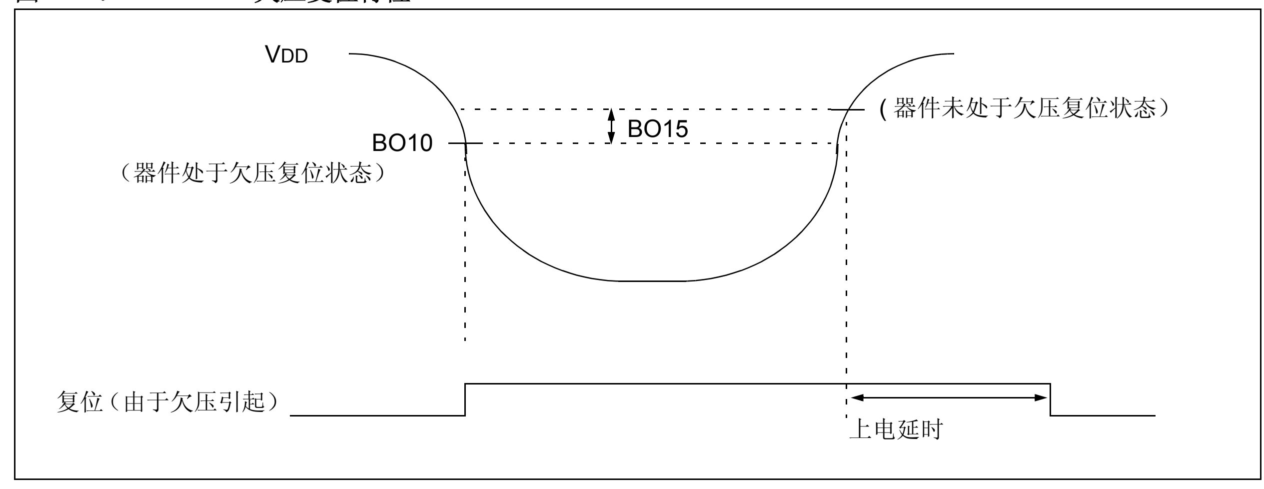


表 22-10： 电气特性：BOR

DC 特性				标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性		最小值	典型值 [1]	最大值	单位	条件
BO10	VBOR	当 VDD 从高电平翻转至低电平时的 BOR 电压 [2]	BORV = 11[3]	—	—	—	V	不在工作电压范围内
			BORV = 10	2.6	—	2.71	V	
			BORV = 01	4.1	—	4.4	V	
			BORV = 00	4.58	—	4.73	V	
BO15	VBHYS			—	5	—	mV	

注 1： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。
2： 这些参数为特性值，但生产时未经测试。
3： 00 值不在可用工作电压范围内。

表 22-11： DC 特性：程序存储器和 EEPROM

DC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值 [1]	最大值	单位	条件
		数据 EEPROM 存储器 [2]					
D120	ED	字节耐擦写能力	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C
D121	VDRW	读 / 写操作时的 VDD	VMIN	—	5.5	V	使用 EECON 读 / 写 VMIN = 最小工作电压
D122	TDEW	擦 / 写周期时间	—	2	—	ms	
D123	TRETD	特性保持时间	40	100	—	年	假定未违反其他规范
D124	IDEW	编程时的 IDD	—	10	30	mA	行擦除
		闪存程序存储器 [2]					
D130	EP	单元耐擦写能力	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C
D131	VPR	读操作时的 VDD	VMIN	—	5.5	V	VMIN = 最小工作电压
D132	VEB	块擦除时的 VDD	4.5	—	5.5	V	
D133	VPEW	擦 / 写操作时的 VDD	3.0	—	5.5	V	
D134	TPEW	擦 / 写周期时间	—	2	—	ms	
D135	TRETD	特性保持时间	40	100	—	年	假定未违反其他规范
D136	TEB	ICSP 块擦除时间	—	4	—	ms	
D137	IPEW	编程时的 IDD	—	10	30	mA	行擦除
D138	IEB	编程时的 IDD	—	10	30	mA	块擦除

注 1： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。
2： 这些参数为特性值，但生产时未经测试。

22.2 AC 特性和时序参数

本节包含的信息说明了 dsPIC30F 系列器件的 AC 特性和时序参数。

表 22-12： 温度和电压规范——AC

AC 特性	标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度 工业级：-40°C ≤ T_A ≤ +85°C 扩展级：-40°C ≤ T_A ≤ +125°C 工作电压 V_{DD} 范围如 DC 规范第 **22.1** 节 **"DC 特性"** 中所介绍。

图 22-2： 器件时序规范的负载条件

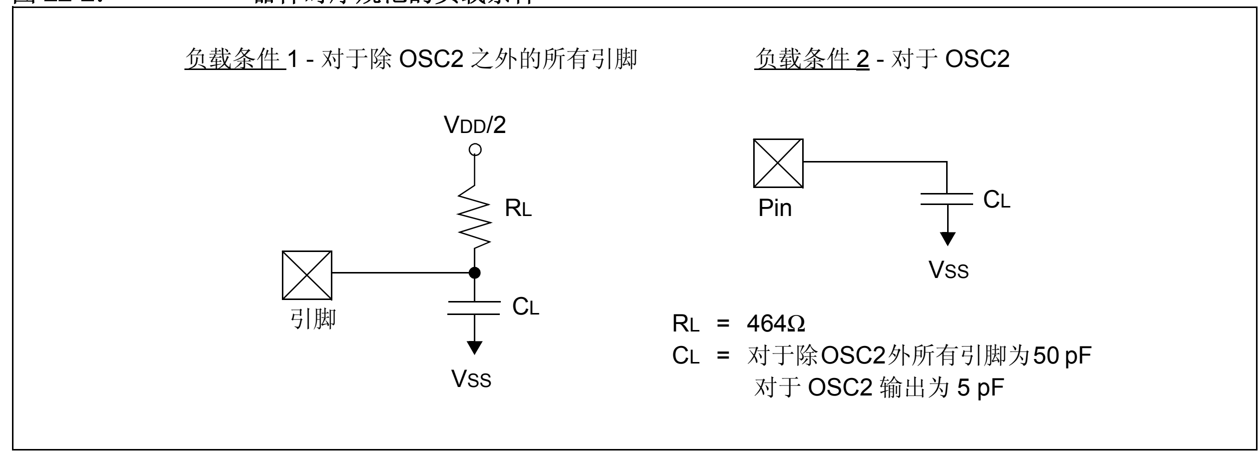


图 22-3： 外部时钟时序

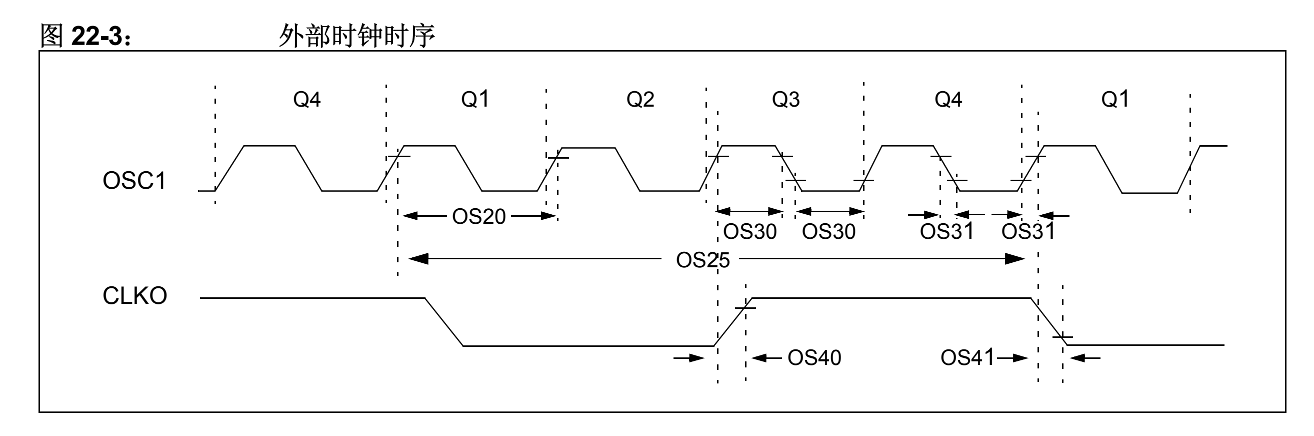


表 22-13： 外部时钟时序要求

AC 特性			标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
OS10	FOSC	外部 CLKI 频率 (2) （外部时钟仅允许运行于 EC 模式）	DC 4 4 4	— — — —	40 10 10 7.5	MHz MHz MHz MHz	EC EC ，且 4x PLL EC ，且 8x PLL EC ，且 16x PLL
		振荡器频率 (2)	DC 0.4 4 4 4 4 10 31 — —	— — — — — — — — 7.37 512	4 4 10 10 10 7.5 25 33 — —	MHz MHz MHz MHz MHz MHz MHz kHz MHz kHz	RC XTL XT XT ，且 4x PLL XT ，且 8x PLL XT ，且 16x PLL HS LP 内部 FRC 内部 LPRC
OS20	TOSC	TOSC = 1/FOSC	—	—	—	—	关于 FOSC 值，参见参数 OS10
OS25	TCY	指令周期时间 (2)(3)	33	—	DC	ns	参见表 22-16
OS30	TosL, TosH	外部时钟 (2) 输入 （OSC1）高电平或低电平时间	.45 x TOSC	—	—	ns	EC
OS31	TosR, TosF	外部时钟 (2) 输入 （OSC1）上升或下降时间	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 (2)(4)	—	6	10	ns	
OS41	TckF	CLKO 下降时间 (2)(4)	—	6	10	ns	

注 1： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。

2： 这些参数为特性值，但生产时未经测试。

3： 指令周期 （TCY）等于输入振荡器时基周期的 4 倍。所有规定值均基于标准运行条件下，器件执行代码时对应特定振荡器类型的特征数据。超过规定值可导致振荡器运行不稳定，并 / 或使电流消耗超过预期。所有器件在测试 “最小”值时，均在 OSC1/CLKI 引脚接入了外部时钟。当使用外部时钟输入时，所有器件的 “最大”周期时间极限为 “DC”（无时钟）。

4： 测量在 EC 或 ERC 模式下进行。在 OSC2 引脚上测量 CLKO 信号。 CLKO 在 Q1-Q2 周期 （1/2 TCY）中为低电平，在 Q3-Q4 周期 （1/2 TCY）中为高电平。

表 22-14： PLL 时钟时序规范 （VDD = 2.5 至 5.5 V）

AC 特性			标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
OS50	FPLLI	PLL 输入频率范围 (2)	4	—	10	MHz	EC，带 4x PLL
			4	—	10	MHz	EC，带 8x PLL
			4	—	7.5(3)	MHz	EC，带 16x PLL
			4	—	10	MHz	XT，带 4x PLL
			4	—	10	MHz	XT，带 8x PLL
			4	—	7.5(3)	MHz	XT，带 16x PLL
OS51	FSYS	片内 PLL 输出 (2)	16	—	120	MHz	EC 和 XT 模式，且 PLL 使能
OS52	TLOC	PLL 启动时间 （锁定时间）	—	20	50	µs	

注 1： 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。
3： 受器件工作频率范围的限制。

表 22-15： PLL 抖动

AC 特性		标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C					
参数编号	特性	最小值	典型值 (1)	最大值	单位	条件	
OS61	x4 PLL	—	0.251	0.413	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0 至 3.6V
		—	0.251	0.413	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0 至 3.6V
		—	0.256	0.47	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5 至 5.5V
		—	0.256	0.47	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V
	x8 PLL	—	0.355	0.584	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0 至 3.6V
		—	0.355	0.584	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0 至 3.6V
		—	0.362	0.664	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5 至 5.5V
		—	0.362	0.664	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V
	x16 PLL	—	0.67	0.92	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0 至 3.6V
		—	0.632	0.956	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5 至 5.5V
		—	0.632	0.956	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V

注 1： 这些参数为特性值，但生产时未经测试。

表 22-16： 内部时钟时序示例

时钟振荡器模式	FOSC（MHz）(1)	TCY（μs）(2)	MIPS(3) 无 PLL	MIPS(3) PLL x4	MIPS(3) PLL x8	MIPS(3) PLL x16
EC	0.200	20.0	0.05	—	—	—
	4	1.0	1.0	4.0	8.0	16.0
	10	0.4	2.5	10.0	20.0	—
	25	0.16	6.25	—	—	—
XT	4	1.0	1.0	4.0	8.0	16.0
	10	0.4	2.5	10.0	20.0	—

注 1： 假设振荡器后分频器分频比为 1。
2： 指令执行周期时间：TCY = 1 / MIPS。
3： 指令执行频率：MIPS = (FOSC * PLLx) / 4 （因为每个指令周期有 4 个 Q 时钟）。

表 22-17： AC 特性：内部 RC 抖动、精度和漂移

AC 特性		标准工作条件：**2.5 V** 至 **5.5 V**（除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C					
参数编号	特性	最小值	典型值	最大值	单位	条件	
	内部 FRC 抖动 @ FRC 频率 = 7.37 MHz(1)						
OS62	FRC	—	±0.04	±0.16	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
		—	±0.07	±0.23	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5-5.5V
	内部 FRC 精度 @ FRC 频率 = 7.37 MHz(1)						
OS63	FRC	—	—	±1.50	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-5.5V
	内部 FRC 漂移 @ FRC 频率 = 7.37 MHz(1)						
OS64		-0.7	—	0.5	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
		-0.7	—	0.7	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V
		-0.7	—	0.5	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5-5.5V
		-0.7	—	0.7	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5-5.5V

注 1： 在 7.372 MHz ±2%、 25°C 和 5V 条件下进行频率校准。 TUN <3:0> 位可用来补偿温度漂移。
2： FRC 总的偏差可通过将抖动、精度和漂移百分比的绝对值相加来计算。

表 22-18： AC 特性：内部 LPRC 精度

AC 特性		标准工作条件：**2.5 V** 至 **5.5 V**（除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	特性	最小值	典型值	最大值	单位	条件
	LPRC @ 频率 = 512 kHz(1)					
OS65		-35	—	+35	%	—

注 1： LPRC 频率将随 VDD 的变化而变化。

图 22-4： **CLKO 和 I/O 时序特性**

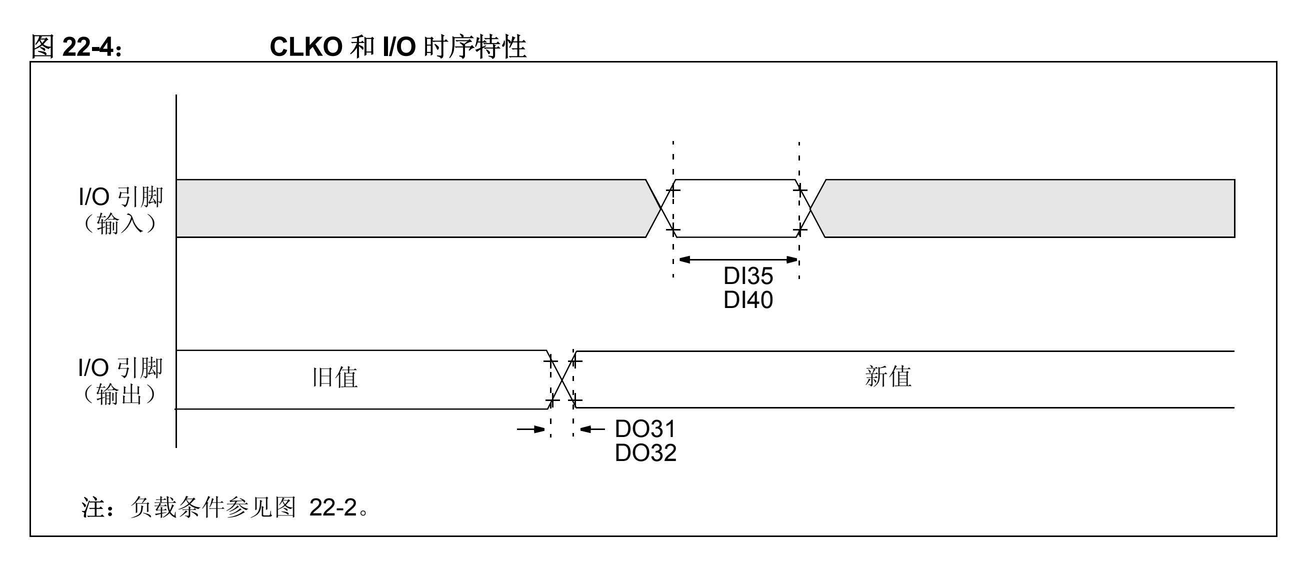
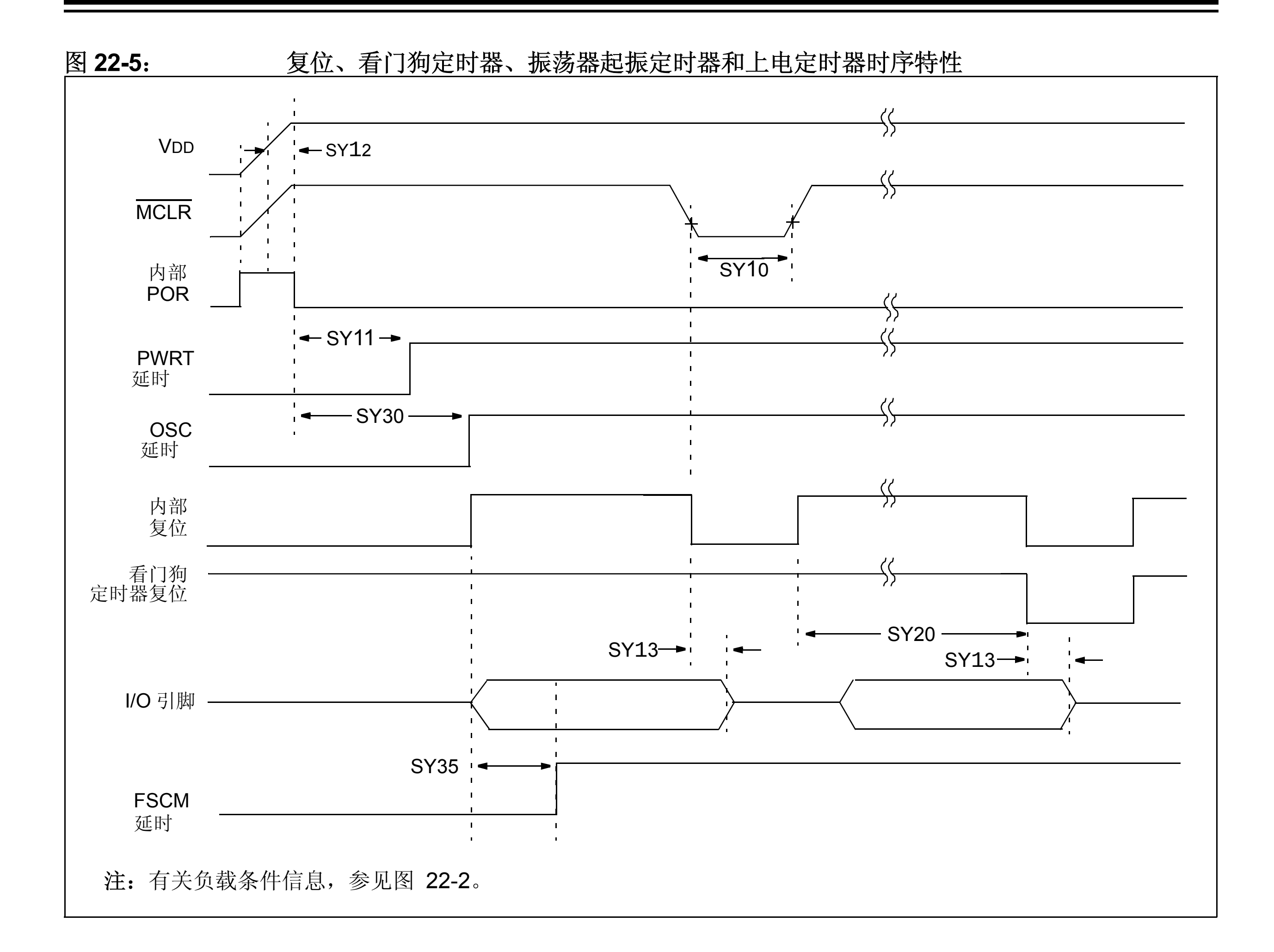


表 22-19： **CLKO 和 I/O 时序要求**

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围　工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参输编号	符号	特性 (1)(2)(3)	最小值	典型值 (4)	最大值	单位	条件
DO31	TIOR	端口输出上升时间	—	7	20	ns	—
DO32	TIOF	端口输出下降时间	—	7	20	ns	—
DI35	TINP	INTx 引脚高电平或低电平时间 （输出）	20	—	—	ns	—
DI40	TRBP	CNx 高电平或低电平时间 （输入）	2 TCY	—	—	—	—

注 1： 这些参数为异步事件，与任何内部时钟边沿无关。
2： 测量是在 RC 模式和 EC 模式下进行的，其中 CLKO 输出为 4 x TOSC。
3： 这些参数为特性值，但生产时未经测试。
4： 除非另外声明，否则 “典型值” 一列中的数据都是在 5V、 25°C 的条件下给出的。

图 22-5： 复位、看门狗定时器、振荡器起振定时器和上电定时器时序特性



注：有关负载条件信息，参见图 22-2。

表 **22-20**： 复位、看门狗定时器、振荡器起振定时器、上电定时器和欠压复位时序要求

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SY10	TmcL	$\overline{\text{MCLR}}$ 脉冲宽度 （低电平）	2	—	—	µs	-40°C 至 +85°C
SY11	TPWRT	上电延时定时器周期	3 12 50	4 16 64	6 22 90	ms	-40°C 至 +85°C 用户可编程
SY12	TPOR	上电复位延时	3	10	30	µs	-40°C 至 +85°C
SY13	TIOZ	$\overline{\text{MCLR}}$ 低电平或看门狗定时器复位引起 I/O 高阻态	—	0.8	1.0	µs	
SY20	TWDT1 TWDT2	看门狗定时器超时周期（无预分频器）	1.4 1.4	2.1 2.1	2.8 2.8	ms ms	3.3V， -40°C 至 +125°C 5.0V， -40°C 至 +125°C
SY25	TBOR	欠压复位脉冲宽度 (3)	100	—	—	µs	VDD ≤ VBOR （D034）
SY30	TOST	振荡器起振定时器周期	—	1024 TOSC	—	—	TOSC = OSC1 周期
SY35	TFSCM	故障保护时钟监视器延时	—	500	900	µs	-40°C 至 +85°C

注 **1**： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值” 一列中的数据都是在 5V、 25°C 的条件下给出的。

3： 有关 BOR 的信息，参见图 22-1 和表 22-10。

图 22-6： 带隙启动时间特性

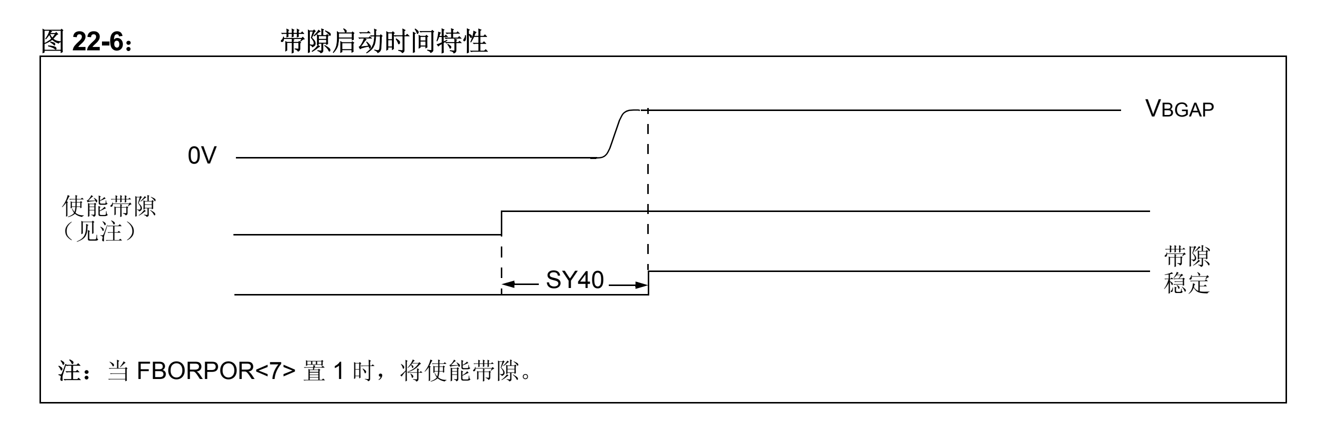
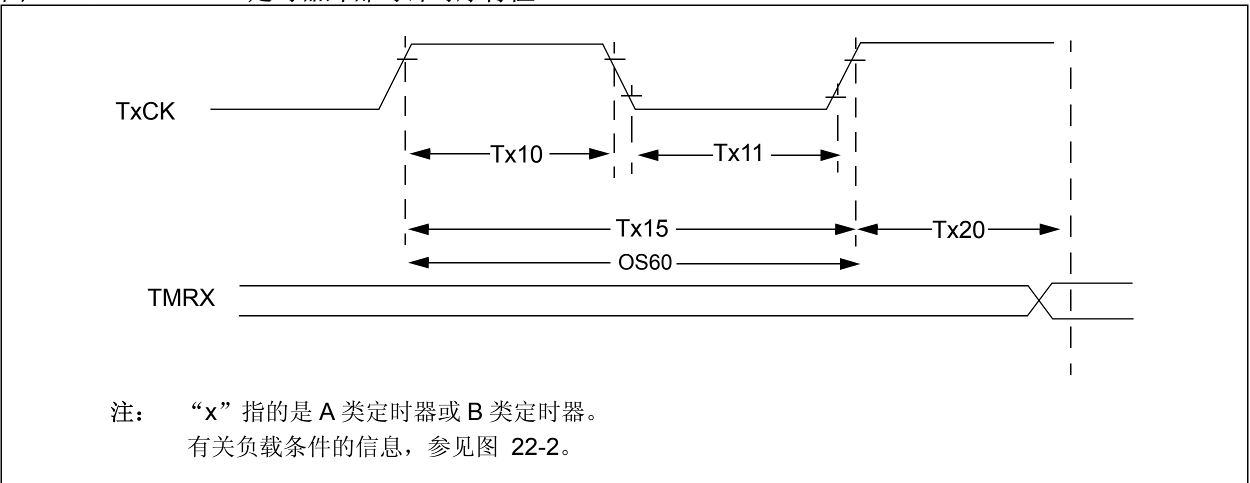


表 22-21： 带隙启动时间要求

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	**符号**	**特性 (1)**	**最小值**	**典型值 (2)**	**最大值**	**单位**	**条件**
SY40	TBGAP	带隙启动时间	—	40	65	µs	定义为带隙功能使能和带隙参考电压稳定之间的时间。 RCON<13> 状态位

注 **1：** 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。

图 **22-7**： 定时器外部时钟时序特性



注： “x”指的是 A 类定时器或 B 类定时器。
有关负载条件的信息，参见图 22-2。

表 **22-22**： **TIMER1** 外部时钟时序要求

AC 特性				标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TA10	TTXH	TxCK 高电平时间	同步， 无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TA15
			同步， 带预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA11	TTXL	TxCK 低电平时间	同步， 无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TA15
			同步， 带预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA15	TTXP	TxCK 输入周期	同步， 无预分频器	TCY + 10	—	—	ns	
			同步， 带预分频器	20 ns 或 (TCY + 40)/N 中的较大值	—	—	—	N = 预分频值 （1, 8, 64, 256）
			异步	20	—	—	ns	
OS60	Ft1	SOSC1/T1CK 振荡器输入频率范围（通过置 1 TCS 位（T1CON 的 bit 1）使能振荡器		DC	—	50	kHz	
TA20	TCKEXTMRL	从外部 TxCK 时钟边沿至定时器递增之间的延时		0.5 TCY		1.5 TCY	—	

表 22-23： **TIMER2 外部时钟时序要求**

AC 特性				标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TB10	TtxH	TxCK 高电平时间	同步，无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TB15
			同步，带预分频器	10	—	—	ns	
TB11	TtxL	TxCK 低电平时间	同步，无预分频器	0.5 TCY + 20	—	—	ns	也必须满足参数 TB15
			同步，带预分频器	10	—	—	ns	
TB15	TtxP	TxCK 输入周期	同步，无预分频器	TCY + 10	—	—	ns	N = 预分频值（1, 8, 64, 256）
			同步，带预分频器	20 ns 或（TCY + 40）/N 两者中的较大值				
TB20	TCKEXTMRL	从外部 TxCK 时钟边沿至定时器递增之间的延时		0.5 TCY		1.5 TCY	—	

表 22-24： **TIMER3 外部时钟时序要求**

AC 特性				标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TC10	TtxH	TxCK 高电平时间	同步	0.5 TCY + 20	—	—	ns	也必须满足参数 TC15
TC11	TtxL	TxCK 低电平时间	同步	0.5 TCY + 20	—	—	ns	也必须满足参数 TC15
TC15	TtxP	TxCK 输入周期	同步，无预分频器	TCY + 10	—	—	ns	N = 预分频值（1, 8, 64, 256）
			同步，带预分频器	20 ns 或（TCY + 40）/N 两者中的较大值				
TC20	TCKEXTMRL	从外部 TxCK 时钟边沿至定时器递增之间的延时		0.5 TCY		1.5 TCY	—	

图 22-8： **TIMERQ （QEI 模块）外部时钟时序特性**

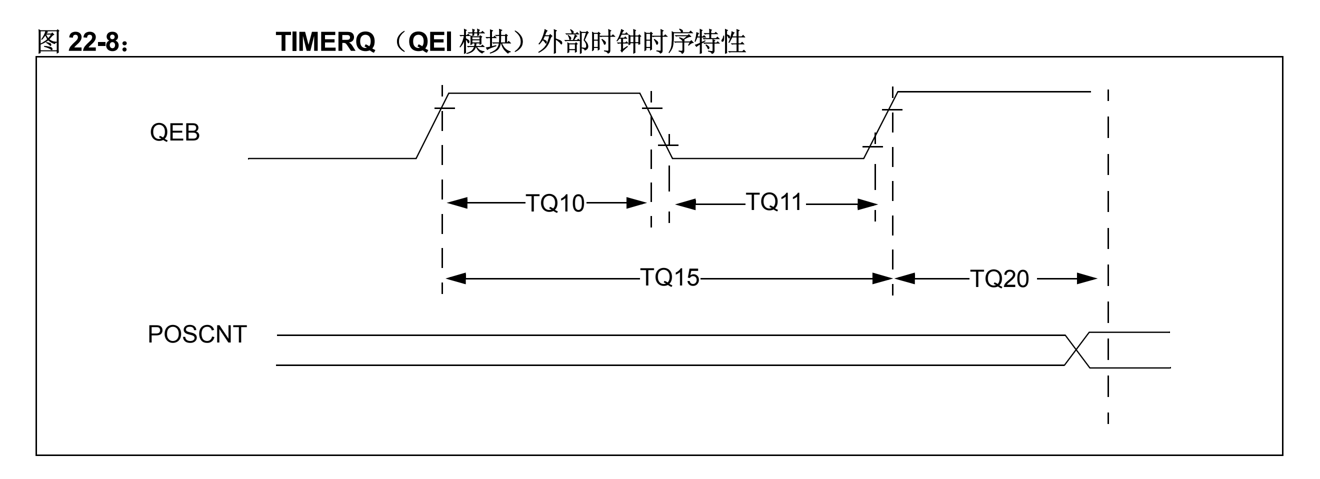
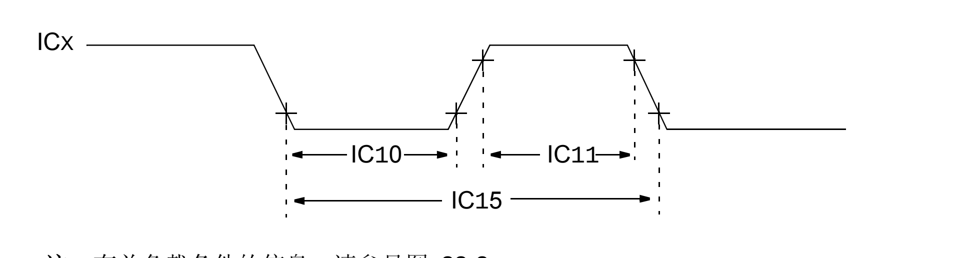


表 22-25： **QEI 模块外部时钟时序要求**

AC 特性				标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 [1]		最小值	典型值	最大值	单位	条件
TQ10	TtQH	TQCK 高电平时间	同步，带预分频器	TCY + 20		—	ns	也必须满足参数 TQ15
TQ11	TtQL	TQCK 低电平时间	同步，带预分频器	TCY + 20		—	ns	也必须满足参数 TQ15
TQ15	TtQP	TQCP 输入周期	同步，带预分频器	2 * TCY + 40		—	ns	—
TQ20	TCKEXTMRL	从外部 TxCK 时钟边沿至定时器递增之间的延时		0.5 TCY		1.5 TCY	ns	—

注 1： 这些参数为特性值，但生产时未经测试。

图 22-9： 输入捕捉 （CAPx）时序特性



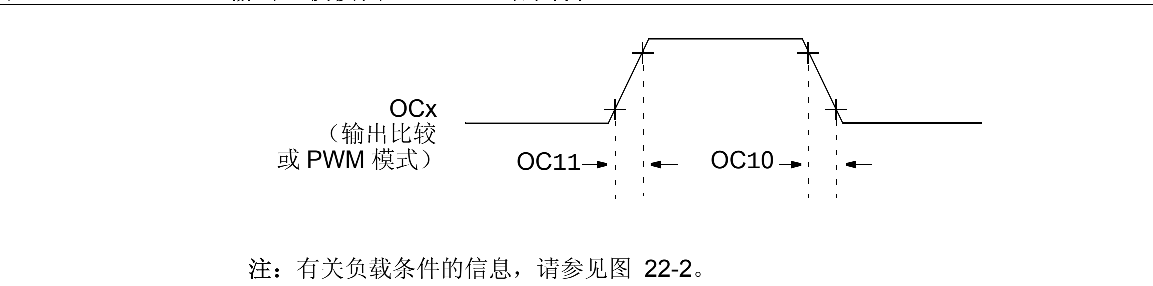
注：有关负载条件的信息，请参见图 22-2。

表 22-26： 输入捕捉时序要求

AC 特性				标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C			
参数编号	符号	特性 [(1)]		最小值	最大值	单位	条件
IC10	TccL	ICx 输入低电平时间	无预分频器	0.5 TCY + 20	—	ns	
			带预分频器	10	—	ns	
IC11	TccH	ICx 输入高电平时间	无预分频器	0.5 TCY + 20	—	ns	
			带预分频器	10	—	ns	
IC15	TccP	ICx 输入周期		(2 TCY + 40)/N	—	ns	N = 预分频值（1, 4, 16）

注 1： 这些参数为特性值，但生产时未经测试。

图 22-10： 输出比较模块 （OCx）时序特性



注：有关负载条件的信息，请参见图 22-2。

表 22-27： 输出比较模块时序要求

AC 特性			标准工作条件：2.5V 至 5.5V （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 [(1)]	最小值	典型值 [(2)]	最大值	单位	条件
OC10	TccF	OCx 输出下降时间	—	—	—	ns	见参数 D032
OC11	TccR	OCx 输出上升时间	—	—	—	ns	见参数 D031

注 1： 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计参考，未经测试。

图 **22-11**： 输出比较 **/PWM** 模块时序特性

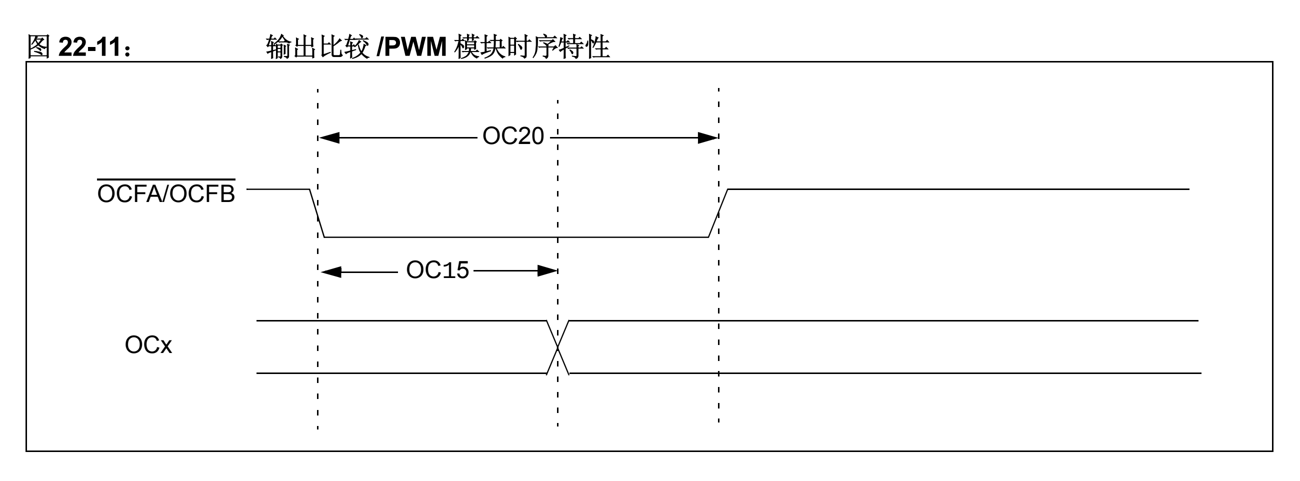


表 **22-28**： 简单输出比较 **/PWM** 模式时序要求

AC 特性				标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C			
参数编号	符号	特性 [(1)]	最小值	典型值 [(2)]	最大值	单位	条件
OC15	TFD	故障输入至 PWM I/O 发生变化的时间	—	—	50	ns	—
OC20	TFLT	故障输入脉冲宽度	50	—	—	ns	—

注 **1**： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值” 一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计时参考，未经测试。

图 22-12： 电机控制 **PWM** 模块故障时序特性

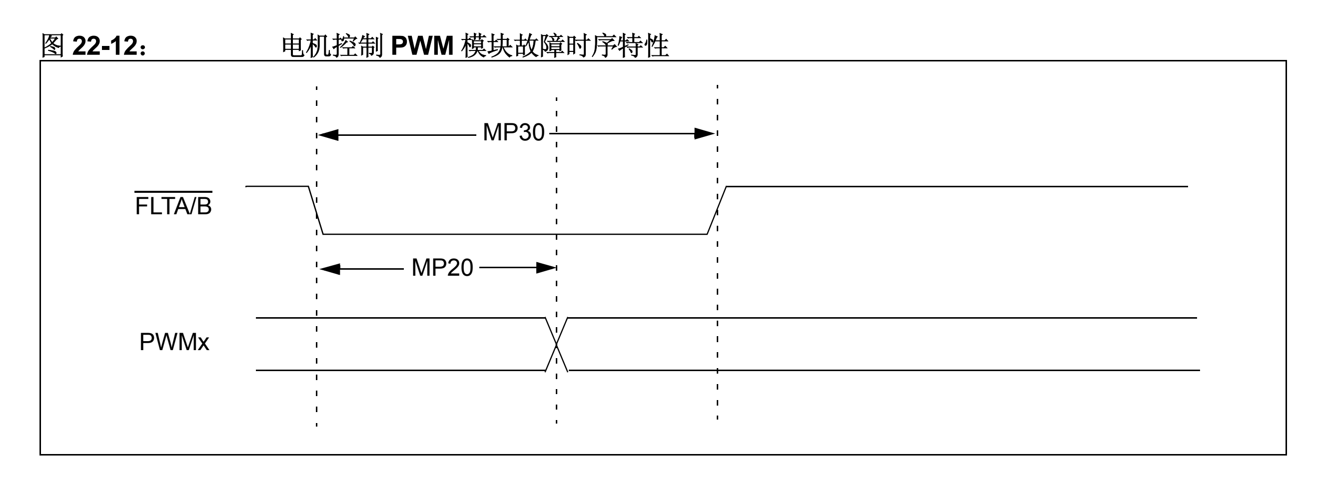


图 22-13： 电机控制 **PWM** 模块时序特性

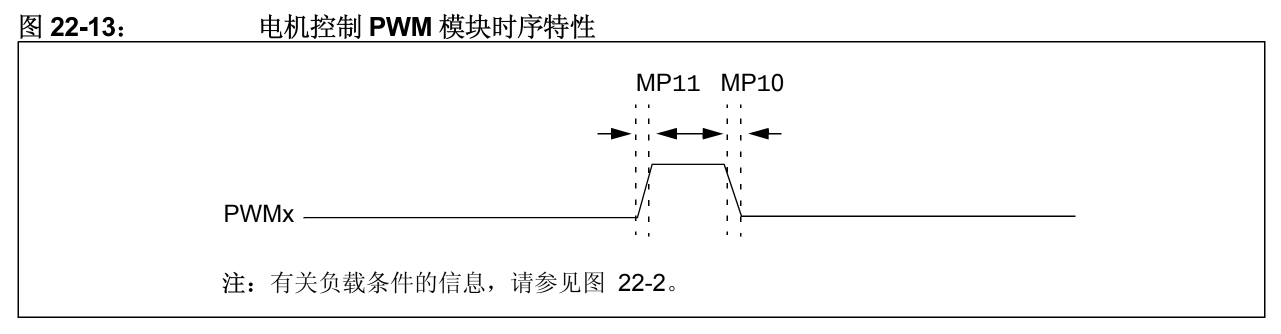


表 22-29： 电机控制 **PWM** 模块时序要求

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
MP10	TFPWM	PWM 输出下降时间	—	—	—	ns	见参数 D032
MP11	TRPWM	PWM 输出上升时间	—	—	—	ns	见参数 D031
MP20	TFD	故障输入↓至 PWMI/O 变化	—	—	50	ns	—
MP30	TFH	最小脉冲宽度	50	—	—	ns	—

注 **1**： 这些参数为特性值，但生产时未经测试。

2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计时参考，未经测试。

图 22-14： QEA/QEB 输入特性

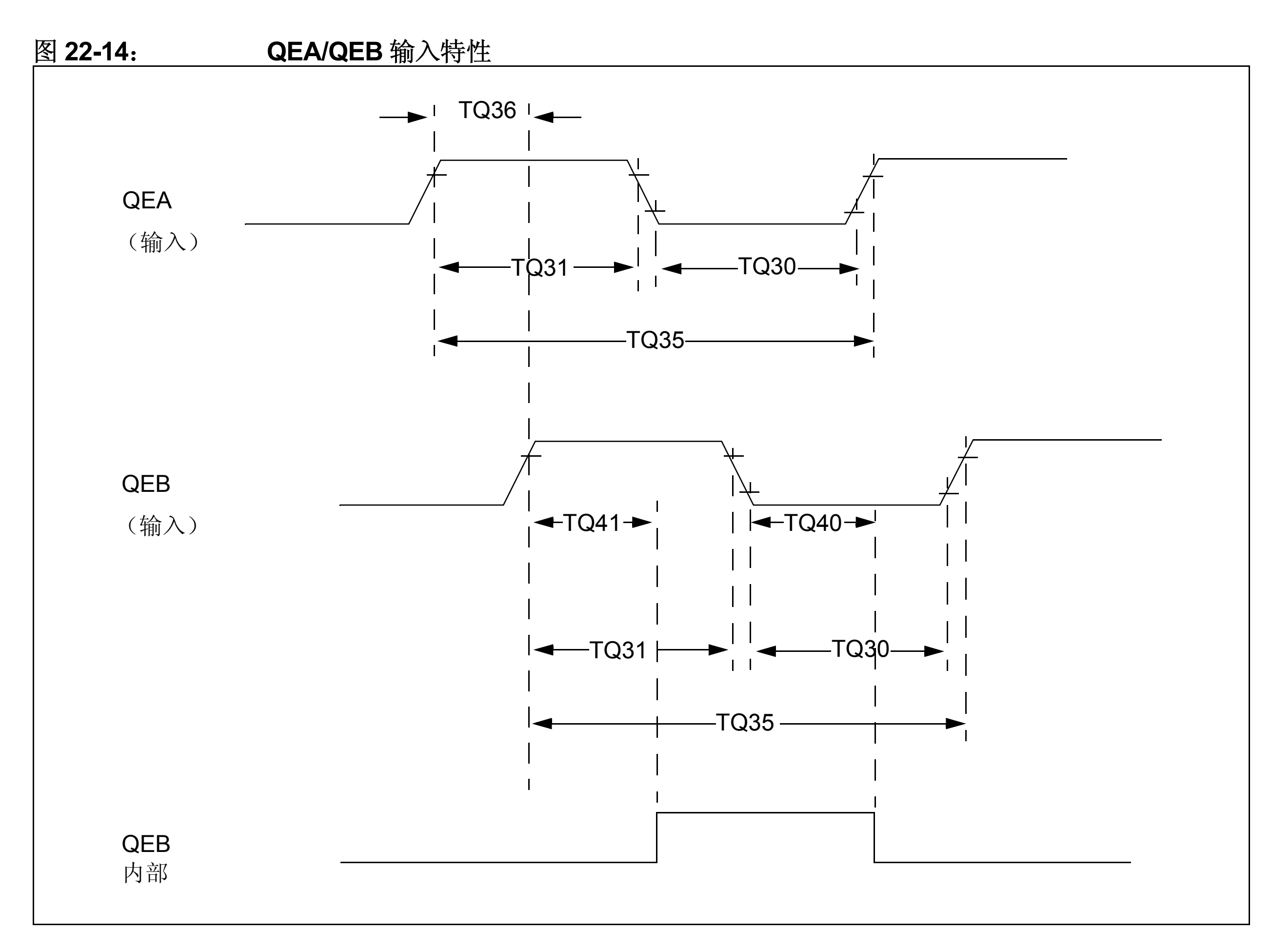


表 22-30： 正交解码器时序要求

AC 特性			标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度范围　工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C			
参数编号	符号	特性 (1)	典型值 (2)	最大值	单位	条件
TQ30	TQUL	正交输入低电平时间	6 TCY	—	ns	—
TQ31	TQUH	正交输入高电平时间	6 TCY	—	ns	—
TQ35	TQUIN	正交输入周期	12 TCY	—	ns	—
TQ36	TQUP	正交相周期	3 TCY	—	ns	—
TQ40	TQUFL	数字滤波器确认低电平的时间	3 * N * TCY	—	ns	N = 1、 2、 4、 16、 32、 64、 128 和 256（注 **2**）
TQ41	TQUFH	数字滤波器确认高电平的时间	3 * N * TCY	—	ns	N = 1、 2、 4、 16、 32、 64、 128 和 256（注 **2**）

注 **1**： 这些参数为特性值，但生产时未经测试。
2： N = 索引通道数字滤波器时钟分频选择位。请参见 《dsPIC30F 系列参考手册》（DS70046E_CN）的第 **16** 章 “正交编码器接口 （**QEI**）”。

图 22-15: QEI 模块索引脉冲时序特性

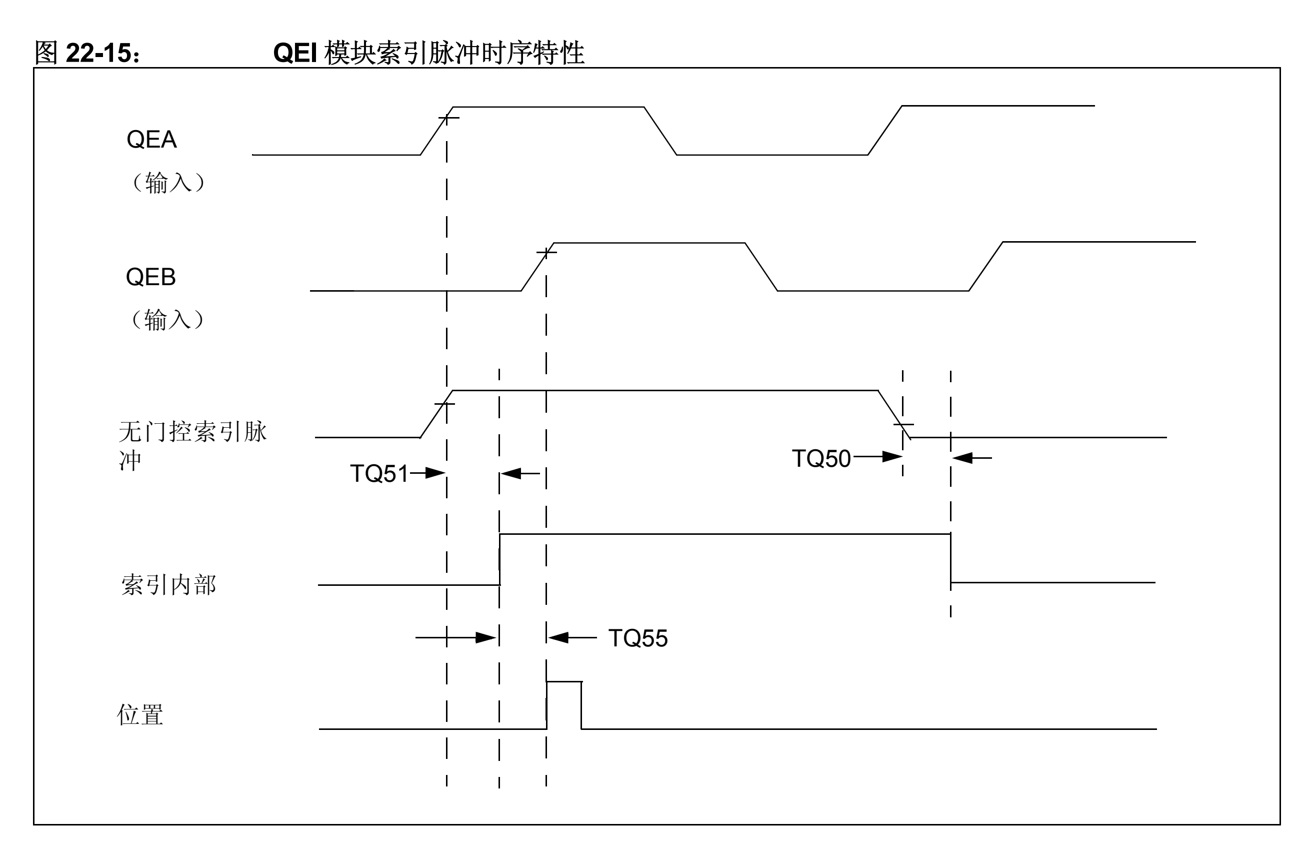


表 22-31: QEI 索引脉冲时序要求

AC 特性			标准工作条件：2.5V 至 5.5V（除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C			
参数编号	符号	特性 (1)	典型值	最大值	单位	条件
TQ50	TqIL	数字滤波器确认低电平的时间	3 * N * TCY	—	ns	N = 1、 2、 4、 16、 32、 64、 128 和 256（注 2）
TQ51	TqiH	数字滤波器确认高电平的时间	3 * N * TCY	—	ns	N = 1、 2、 4、 16、 32、 64、 128 和 256（注 2）
TQ55	Tqidxr	索引脉冲确认至位置计数器复位）（无门控索引）	3 TCY	—	ns	—

注 1： 这些参数为特性值，但生产时未经测试。

2： 索引脉冲与 QEA 和 QEB 对齐仅适用于位置计数器复位时序和正向旋转方式 （QEA 超前 QEB）。同样的时序适用于反向旋转的情形 （QEA 滞后于 QEB），但索引脉冲确认出现在下降沿。

图 22-16： SPI 模块主模式 （CKE = 0）时序特性

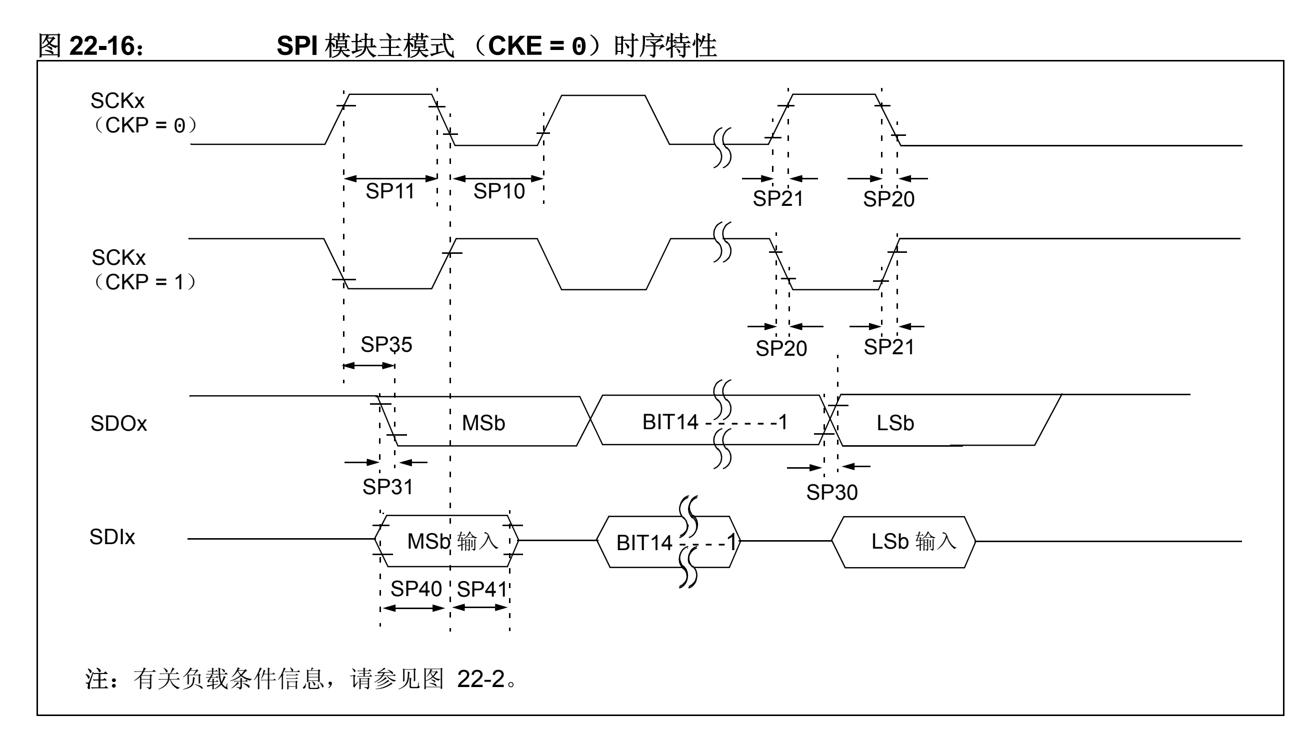
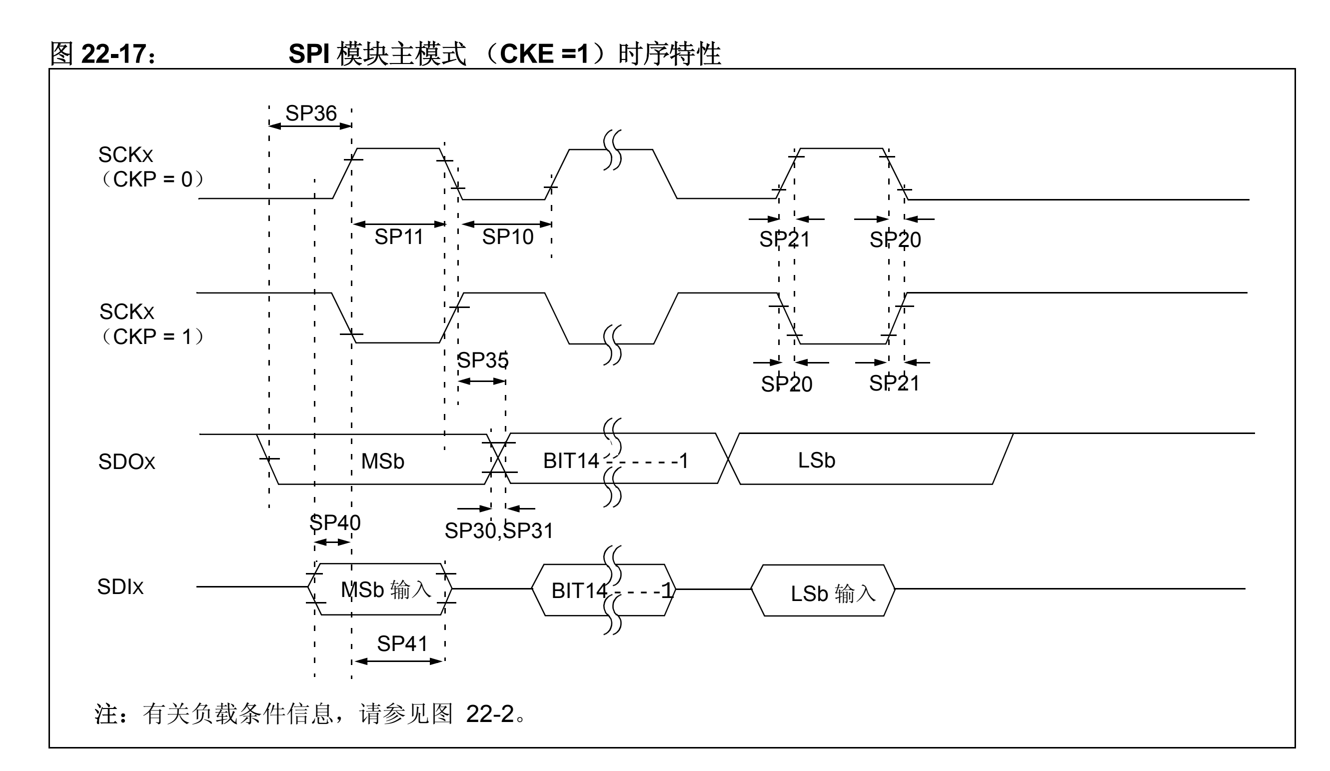


表 22-32： SPI 主模式 （CKE = 0）时序要求

AC 特性			标准工作条件：2.5V 至 5.5V（除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 (3)	TCY / 2	—	—	ns	—
SP11	TscH	SCKx 输出高电平时间 (3)	TCY / 2	—	—	ns	—
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 D032
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx 边沿后的 SDOx 数据输出有效时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到SCKx边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到SCKx边沿的保持时间	20	—	—	ns	—

注 1： 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计时参考，未经测试。
3： SCK 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。
4： 假定所有 SPI 引脚上为 50 pF 负载。

图 22-17： SPI 模块主模式 （CKE =1）时序特性



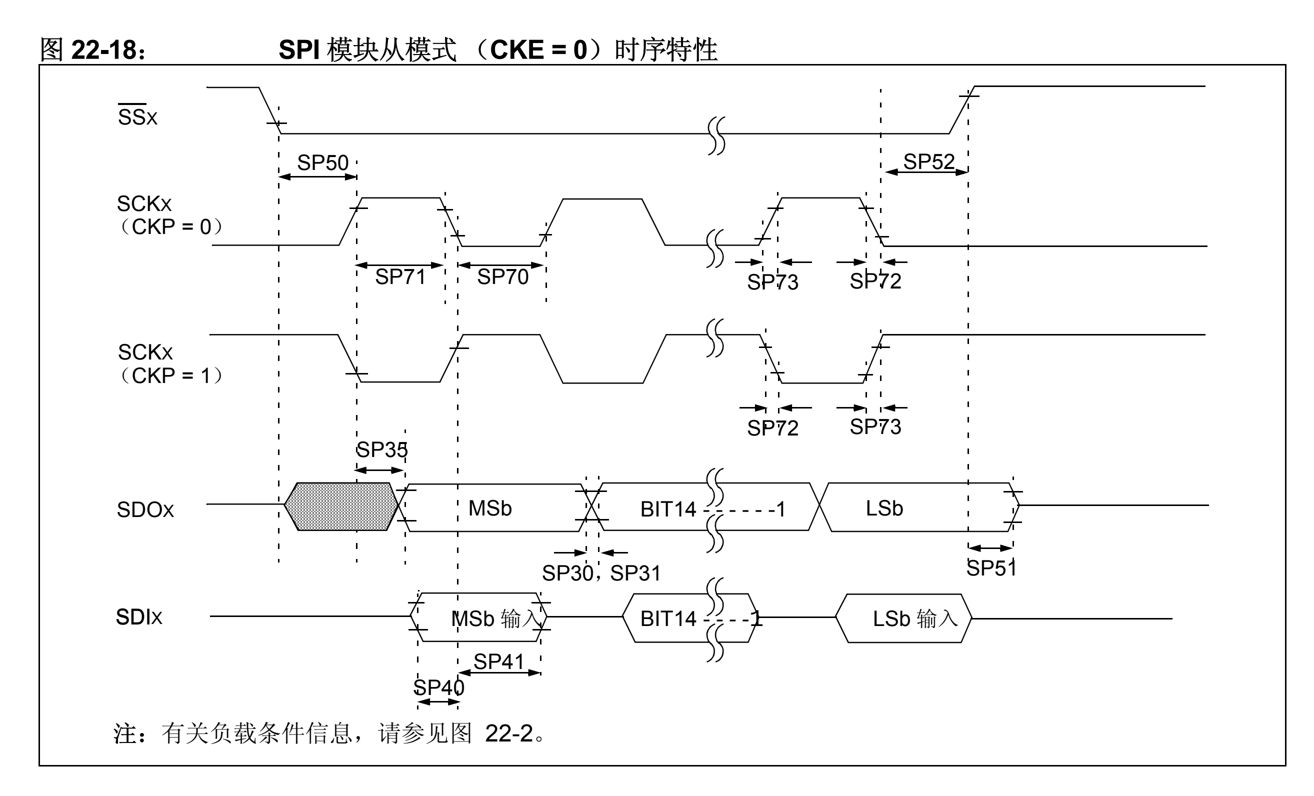
注：有关负载条件信息，请参见图 22-2。

表 22-33： SPI 模块主模式 （CKE = 1）时序要求

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 (3)	TCY / 2	—	—	ns	—
SP11	TscH	SCKx 输出高电平时间 (3)	TCY / 2	—	—	ns	—
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 D032
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx 边沿后的 SDOx 数据输出有效时间	—	—	30	ns	—
SP36	TdoV2sc, TdoV2scL	SDOx 数据输出建立到 SCKx 边沿的时间	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—

注 **1：** 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计时参考未经测试。
3： SCK 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。
4： 假定所有 SPI 引脚上为 50 pF 负载。

图 22-18： SPI 模块从模式 （CKE = 0）时序特性



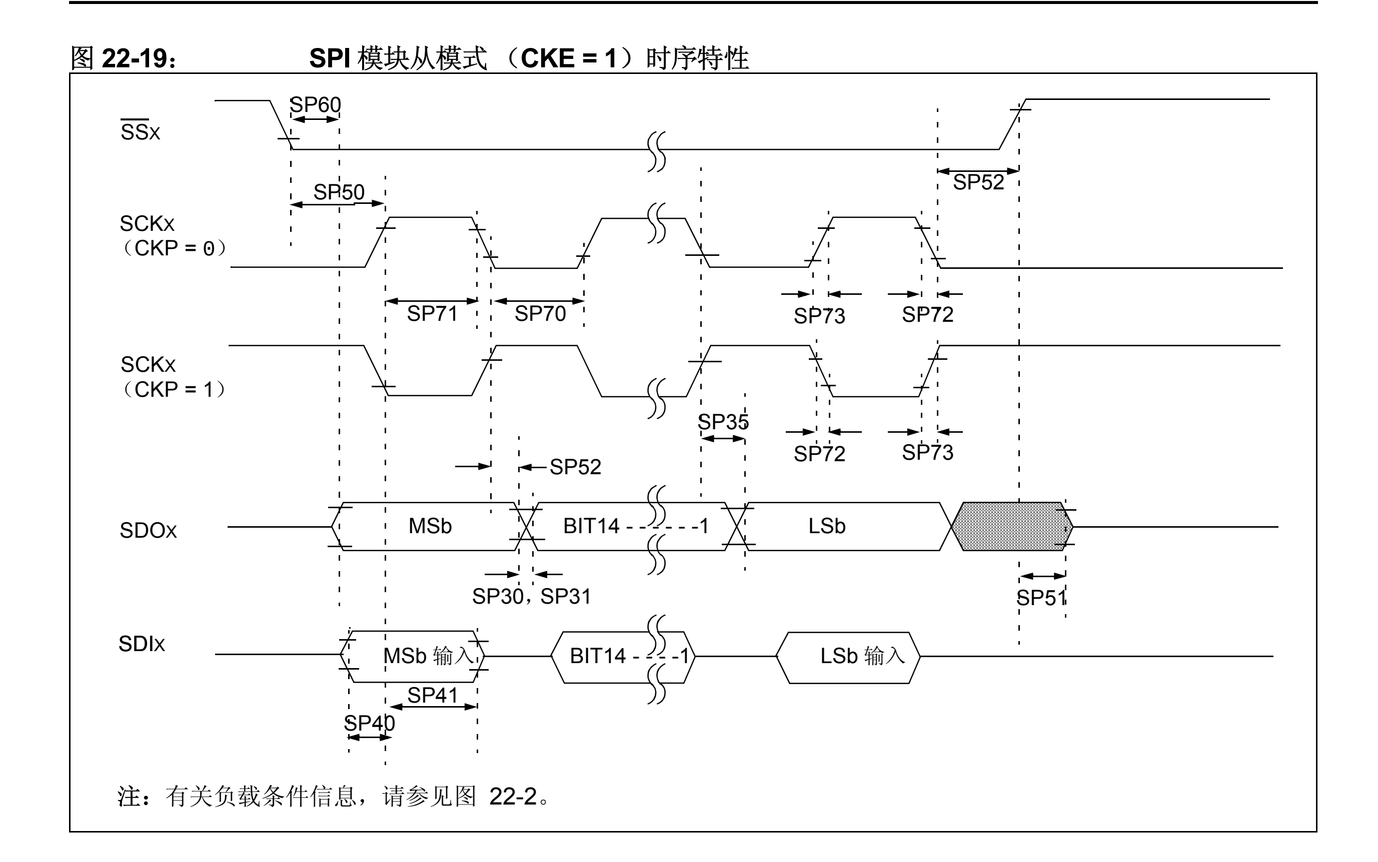
注：有关负载条件信息，请参见图 22-2。

表 22-34： SPI 模块从模式 （CKE = 0）时序要求

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 (3)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx边沿后的SDOx数据输出有效时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SS}$x↓ 至 SCKx↑ 或 SCKx↓ 输入时间	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SS}$x↑ 至 SDOx 输出高阻态的时间 (3)	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	SCK 边沿后的 $\overline{SS}$x	1.5 TCY +40	—	—	ns	—

注 1： 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值”一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计时参考，未经测试。
3： 假定所有 SPI 引脚上为 50 pF 负载。

图 22-19： SPI 模块从模式 （CKE = 1）时序特性



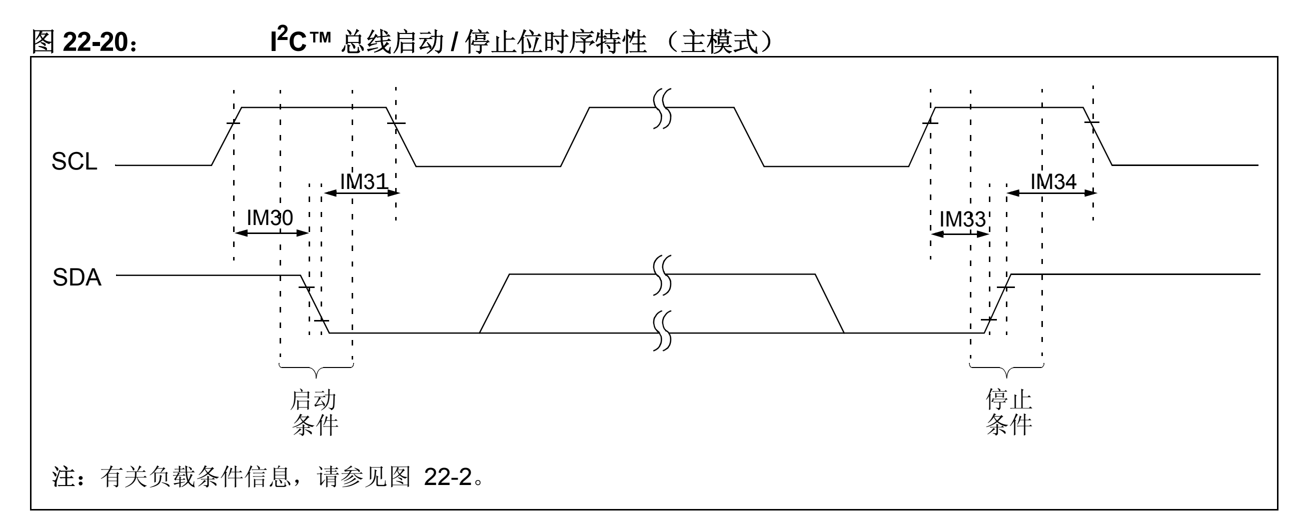
注：有关负载条件信息，请参见图 22-2。

表 22-35： **SPI 模块从模式 （CKE = 1）时序要求**

AC 特性			标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性 [1]	最小值	典型值 [2]	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 [3]	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 [3]	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 [3]	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 [3]	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx 边沿后的 SDOx 数据输出有效时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SS}$x↓至 SCKx↓或 SCKx↑输入时间	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SS}$↑至 SDOx 输出高阻态时间 [4]	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	SCKx 边沿后 $\overline{SS}$x↑	1.5 TCY + 40	—	—	ns	—
SP60	TssL2doV	$\overline{SS}$x 边沿后的 SDOx 数据输出有效时间	—	—	50	ns	—

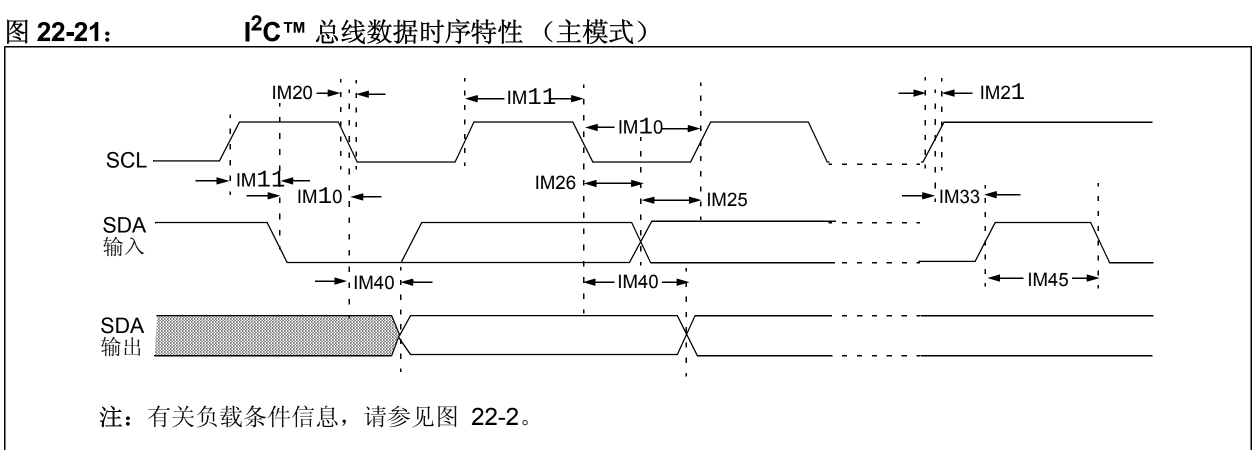
注 1： 这些参数为特性值，但生产时未经测试。
2： 除非另外声明，否则 “典型值” 一列中的数据都是在 5V、 25°C 的条件下给出的。这些参数仅供设计时参考，未经测试。
3： SCK 的最小时钟周期为 100 ns。因此，主模式下产生的时钟不应违反此规范。
4： 假定所有 SPI 引脚上为 50 pF 负载。

图 22-20： I²C™ 总线启动 / 停止位时序特性 （主模式）



注：有关负载条件信息，请参见图 22-2。

图 22-21： I²C™ 总线数据时序特性 （主模式）



注：有关负载条件信息，请参见图 22-2。

表 22-36： I²C™ 总线数据时序要求 （主模式）

AC 特性				标准工作条件：**2.5V 至 5.5V** （除非另外声明） 工作温度范围 工业级：$-40°C \le T_A \le +85°C$ 扩展级：$-40°C \le T_A \le +125°C$			
参数编号	符号	特性		最小值 (1)	最大值	单位	条件
IM10	$T_{LO:SCL}$	时钟低电平时间	100 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	—
			400 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	—
			1 MHz 模式 (2)	T_{CY} / 2 (BRG + 1)	—	μs	—
IM11	$T_{HI:SCL}$	时钟高电平时间	100 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	—
			400 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	—
			1 MHz 模式 (2)	T_{CY} / 2 (BRG + 1)	—	μs	—
IM20	$T_{F:SCL}$	SDA 和 SCL 下降时间	100 kHz 模式	—	300	ns	C_B 规定为 10 至 400 pF
			400 kHz 模式	20 + 0.1 C_B	300	ns	
			1 MHz 模式 (2)	—	100	ns	
IM21	$T_{R:SCL}$	SDA 和 SCL 上升时间	100 kHz 模式	—	1000	ns	C_B 规定为 10 至 400 pF
			400 kHz 模式	20 + 0.1 C_B	300	ns	
			1 MHz 模式 (2)	—	300	ns	
IM25	$T_{SU:DAT}$	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 (2)	TBD	—	ns	
IM26	$T_{HD:DAT}$	数据输入保持时间	100 kHz 模式	0	—	ns	—
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 (2)	TBD	—	ns	
IM30	$T_{SU:STA}$	启动条件建立时间	100 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	仅与重复启动条件相关
			400 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	
			1 MHz 模式 (2)	T_{CY} / 2 (BRG + 1)	—	μs	
IM31	$T_{HD:STA}$	启动条件保持时间	100 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	在此时间之后将产生第一个时钟脉冲
			400 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	
			1 MHz 模式 (2)	T_{CY} / 2 (BRG + 1)	—	μs	
IM33	$T_{SU:STO}$	停止条件建立时间	100 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	—
			400 kHz 模式	T_{CY} / 2 (BRG + 1)	—	μs	
			1 MHz 模式 (2)	T_{CY} / 2 (BRG + 1)	—	μs	
IM34	$T_{HD:STO}$	停止条件保持时间	100 kHz 模式	T_{CY} / 2 (BRG + 1)	—	ns	—
			400 kHz 模式	T_{CY} / 2 (BRG + 1)	—	ns	
			1 MHz 模式 (2)	T_{CY} / 2 (BRG + 1)	—	ns	
IM40	$T_{AA:SCL}$	自时钟边沿到输出有效的时间	100 kHz 模式	—	3500	ns	—
			400 kHz 模式	—	1000	ns	—
			1 MHz 模式 (2)	—	—	ns	—
IM45	$T_{BF:SDA}$	总线空闲时间	100 kHz 模式	4.7	—	μs	在新的发送操作启动之前总线必须保持空闲的时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 (2)	TBD	—	μs	
IM50	C_B	总线容性负载		—	400	pF	

注 **1：** BRG 为 I²C™ 波特率发生器的取值。参见 《dsPIC30F 系列参考手册》（DS70046E_CN）的**第 21 章 “I2C™ 模块”**。

2： 对于所有 I²C 引脚，最大引脚电容 = 10 pF （仅对于 1 MHz 模式）

图 22-22： I^2C™ 总线启动 / 停止位时序特性 （从模式）

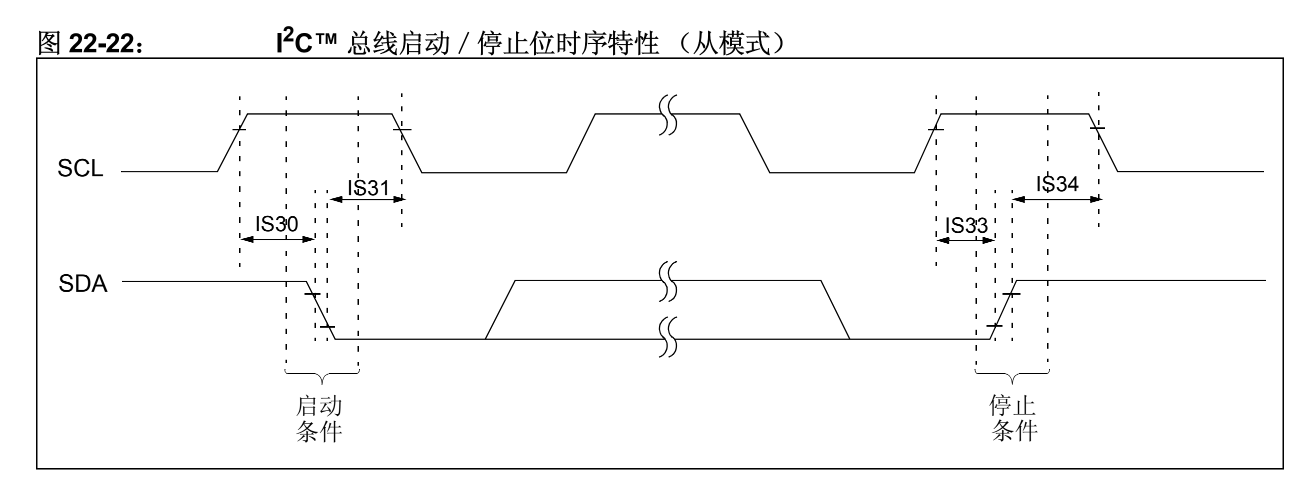


图 22-23： I^2C™ 总线数据时序特性 （从模式）

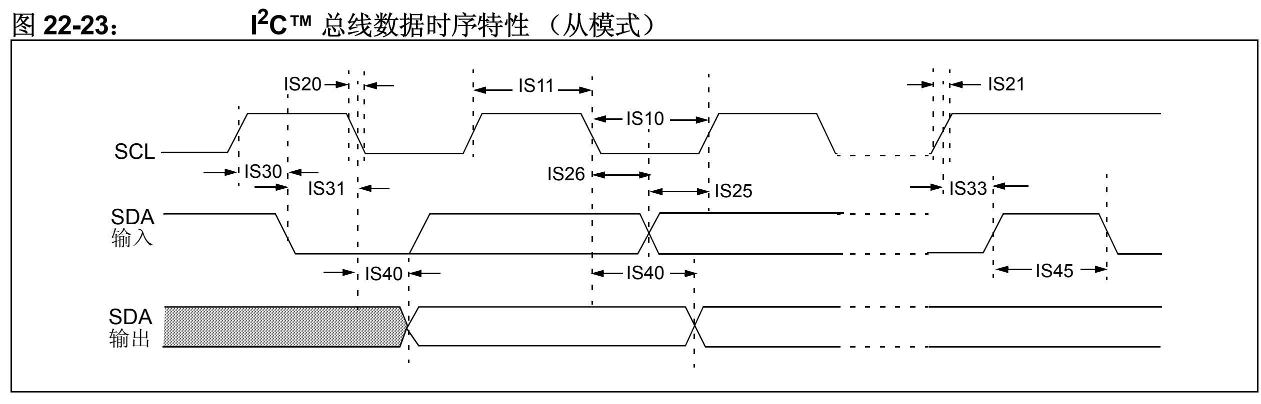


表 22-37： I2C™ 总线数据时序要求 （从模式）

AC 特性		标准工作条件：**2.5V** 至 **5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C					
参数编号 .	符号	特性		最小值	最大值	单位	条件
IS10	TLO:SCL	时钟低电平时间	100 kHz 模式	4.7	—	µs	器件工作频率不能低于 1.5 MHz
			400 kHz 模式	1.3	—	µs	器件工作频率不能低于 10 MHz.
			1 MHz 模式 (1)	0.5	—	µs	—
IS11	THI:SCL	时钟高电平时间	100 kHz 模式	4.0	—	µs	器件工作频率不能低于 1.5 MHz
			400 kHz 模式	0.6	—	µs	器件工作频率不能低于 10 MHz
			1 MHz 模式 (1)	0.5	—	µs	—
IS20	TF:SCL	SDA和SCL下降时间	100 kHz 模式	—	300	ns	CB 规定为 10 至 400 pF
			400 kHz 模式	20 + 0.1 CB	300	ns	
			1 MHz 模式 (1)	—	100	ns	
IS21	TR:SCL	SDA和SCL上升时间	100 kHz 模式	—	1000	ns	CB 规定为 10 至 400 pF
			400 kHz 模式	20 + 0.1 CB	300	ns	
			1 MHz 模式 (1)	—	300	ns	
IS25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 (1)	100	—	ns	
IS26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns	—
			400 kHz 模式	0	0.9	µs	
			1 MHz 模式 (1)	0	0.3	µs	
IS30	TSU:STA	启动条件建立时间	100 kHz 模式	4.7	—	µs	仅与重复启动条件相关
			400 kHz 模式	0.6	—	µs	
			1 MHz 模式 (1)	0.25	—	µs	
IS31	THD:STA	启动条件保持时间	100 kHz 模式	4.0	—	µs	在此时间之后将产生第一个时钟脉冲
			400 kHz 模式	0.6	—	µs	
			1 MHz 模式 (1)	0.25	—	µs	
IS33	TSU:STO	停止条件建立时间	100 kHz 模式	4.7	—	µs	—
			400 kHz 模式	0.6	—	µs	
			1 MHz 模式 (1)	0.6	—	µs	
IS34	THD:STO	停止条件保持时间	100 kHz 模式	4000	—	ns	—
			400 kHz 模式	600	—	ns	
			1 MHz 模式 (1)	250		ns	
IS40	TAA:SCL	自时钟边沿到输出有效的时间	100 kHz 模式	0	3500	ns	—
			400 kHz 模式	0	1000	ns	
			1 MHz 模式 (1)	0	350	ns	
IS45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	µs	在新的发送操作启动之前总线必须保持空闲的时间
			400 kHz 模式	1.3	—	µs	
			1 MHz 模式 (1)	0.5	—	µs	
IS50	CB	总线容性负载		—	400	pF	—

注 1： 对于所有 I2C™ 引脚，最大引脚电容 = 10 pF （仅对于 1 MHz 模式）。

表 22-38： 10 位高速 A/D 模块规范

AC 特性			标准工作条件：2.7V 至 5.5V （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
			器件电源				
AD01	AVDD	模块 VDD 电源	VDD - 0.3 和 2.7 中的较大值		VDD + 0.3 和 5.5 中的较小值	V	—
AD02	AVSS	模块 VSS 电源	Vss - 0.3		VSS + 0.3	V	—
			参考电压源				
AD05	VREFH	参考电压高电平	AVss+2.7		AVDD	V	—
AD06	VREFL	参考电压低电平	AVss		AVDD - 2.7	V	—
AD07	VREF	绝对参考电压	AVss - 0.3		AVDD + 0.3	V	—
AD08	IREF	消耗电流	—	200 .001	300 3	µA µA	A/D 工作 A/D 关闭
			模拟输入				
AD10	VINH-VINL	满量程输入范围	VREFL		VREFH	V	—
AD11	VIN	绝对输入电压	AVSS - 0.3		AVDD + 0.3	V	—
AD12	—	泄露电流	—	±0.001	±0.244	µA	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 5V 电源阻抗 = 5 kΩ
AD13	—	泄露电流	—	±0.001	±0.244	µA	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 3V 信号源阻抗 = 5 kΩ
AD17	RIN	模拟信号源的推荐阻抗	—		5K	Ω	—
			DC 精度				
AD20	Nr	分辨率	10 个数据位			位	—
AD21	INL	积分非线性 (3)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 5V
AD21A	INL	积分非线性 (3)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 3V
AD22	DNL	微分非线性 (3)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 5V
AD22A	DNL	微分非线性 (3)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 3V
AD23	GERR	增益误差 (3)	±1	±5	±6	LSb	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 5V
AD23A	GERR	增益误差 (3)	±1	±5	±6	LSb	VINL = AVSS = VREFL = 0V， AVDD = VREFH = 3V

注 1： 因为采样电容最终将无法保持电荷，因此低于 10 kHz 的时钟频率可能影响线性性能，尤其是在温度较高时。

2： A/D 转换结果将不会随着输入电压的增加而减小，而且不会丢失码。

3： 测量是在使用外部 VREF+ 和 VREF- 作为 ADC 参考电压的情况下进行的。

表 22-38： 10 位高速 A/D 模块规范 （续）

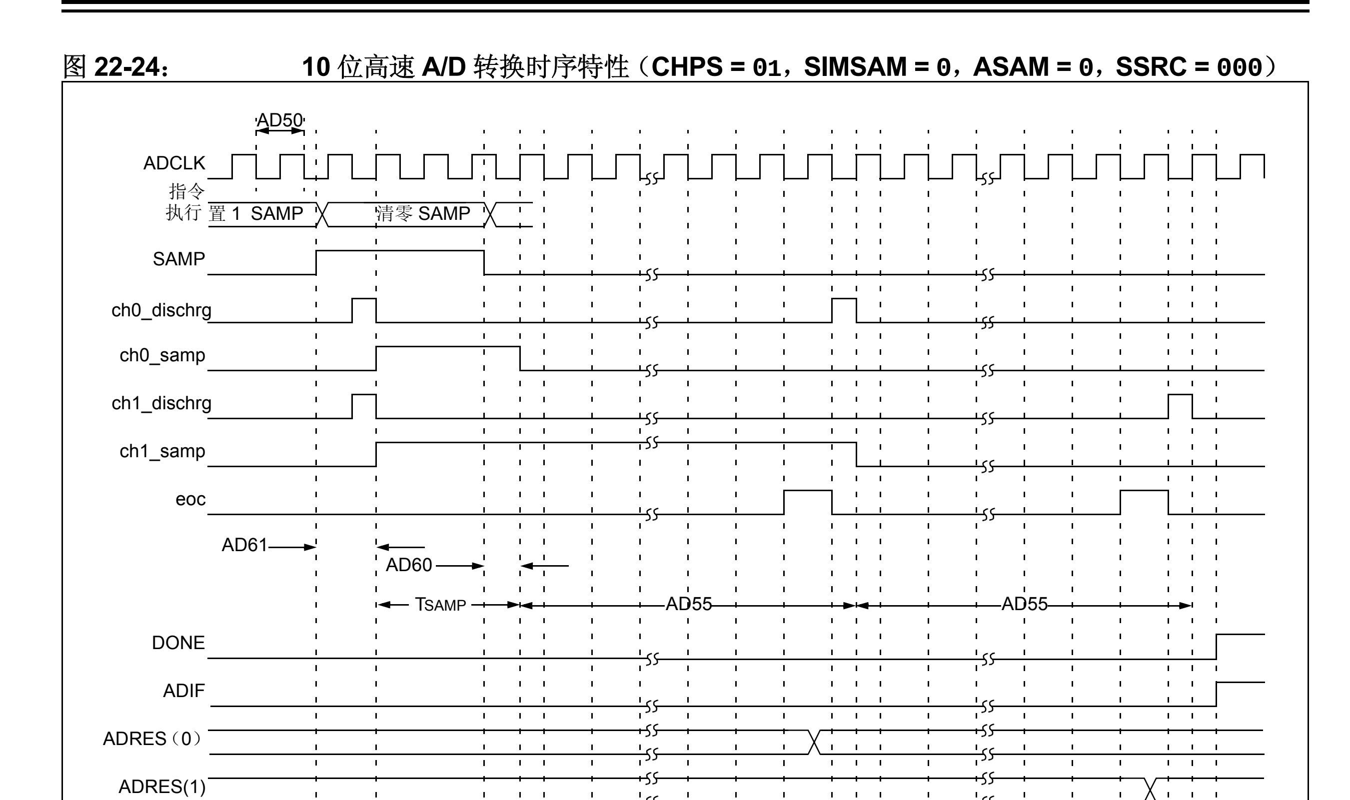
AC 特性			标准工作条件：2.7V 至 5.5V （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
AD24	EOFF	偏移误差	±1	±2	±3	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 5V
AD24A	EOFF	偏移误差	±1	±2	±3	LSb	VINL = AVSS = VREFL = 0V，AVDD = VREFH = 3V
AD25	—	单调性 (2)	—	—	—	—	保证
动态性能							
AD30	THD	总谐波失真	—	-64	-67	dB	—
AD31	SINAD	信号噪声和失真	—	57	58	dB	—
AD32	SFDR	伪空闲动态范围	—	67	71	dB	—
AD33	FNYQ	输入信号带宽	—	—	500	kHz	—
AD34	ENOB	有效位数	9.29	9.41	—	位	—

注 1： 因为采样电容最终将无法保持电荷，因此低于 10 kHz 的时钟频率可能影响线性性能，尤其是在温度较高时。

2： A/D 转换结果将不会随着输入电压的增加而减小，而且不会丢失码。

3： 测量是在使用外部 VREF+ 和 VREF- 作为 ADC 参考电压的情况下进行的。

图 22-24： 10 位高速 A/D 转换时序特性（CHPS = 01，SIMSAM = 0，ASAM = 0，SSRC = 000）



① – 软件置 1 ADCON. SAMP 以启动采样。

② – 有关放电周期 TSAMP 之后的采样启动在第 18.7 节中进行介绍。

③ – 软件清零 ADCON. SAMP 以启动转换。

④ – 采样结束，转换过程启动。

⑤ – 转换 bit 9。

⑥ – 转换 bit 8。

⑧ – 转换 bit 0。

⑨ – 转换结束的一个 TAD。

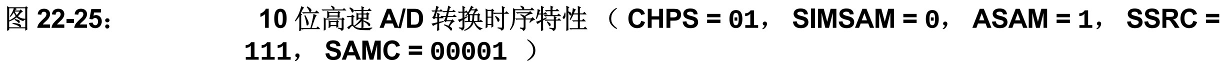


图 22-25： 10 位高速 A/D 转换时序特性 （ CHPS = 01， SIMSAM = 0， ASAM = 1， SSRC = 111， SAMC = 00001 ）

① – 软件置 1 ADCON. ADON 以启动 AD 操作。

② – 采样在放电周期之后启动。
TSAMP 在《dsPIC30F/33F 系列参考手册》（DS70046E_CN）的第 17 章进行介绍。

③ – 转换 bit 9。

④ – 转换 bit 8。

⑤ – 转换 bit 0。

⑥ – 转换结束的一个 TAD。

⑦ – 开始下一个通道的转换。

⑧ – 在由 SAMC 指定的时间内进行采样。
TSAMP 在第 18.7 节介绍。

表 22-39： 10 位高速 A/D 转换时序要求

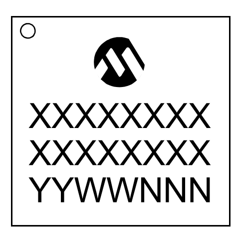
AC 特性			标准工作条件：**2.7V** 至 **5.5V** （除非另外声明） 工作温度范围 工业级：-40°C ≤ TA ≤ +85°C 扩展级：-40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
时钟参数							
AD50	TAD	A/D 时钟周期	—	84	—	ns	见表 **18-2**(1)
AD51	tRC	A/D 内部 RC 振荡器周期	700	900	1100	ns	—
转换速率							
AD55	tCONV	转换时间	—	12 TAD	—	—	—
AD56	FCNV	吞吐率	—	1.0	—	Msps	见表 **18-1**(1)
AD57	TSAMP	采样时间	—	1 TAD	—	—	见表 **18-1**(1)
时序参数							
AD60	tPCS	从采样启动至转换触发的时间	—	1.0 TAD	—	ns	—
AD61	tPSS	从采样位 （SAMP）置 1 至采样启动的时间	0.5 TAD	—	1.5 TAD	ns	—
AD62	tCSS	转换结束至采样启动 （ASAM = 1）时间	—	0.5 TAD	—	ns	—
AD63	tDPU	从 A/D 关闭到 A/D 开始工作使模拟级稳定的时间	—	20	—	µs	—

注 **1：** 因为采样电容最终将无法保持电荷，因此低于 10 kHz 的时钟频率可能影响线性性能，尤其是在温度较高时。

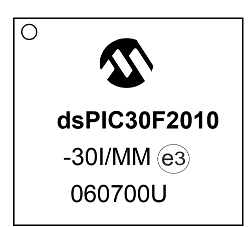
23.0 封装信息

23.1 封装标识信息

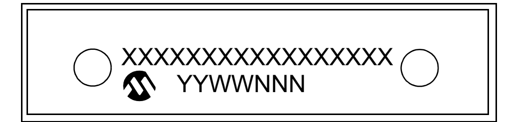
28 引脚 QFN-S



示例



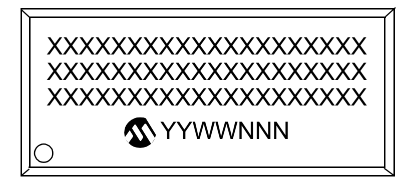
28 引脚 SPDIP（窄条 DIP）



示例



28 引脚 SOIC（.300”）



示例

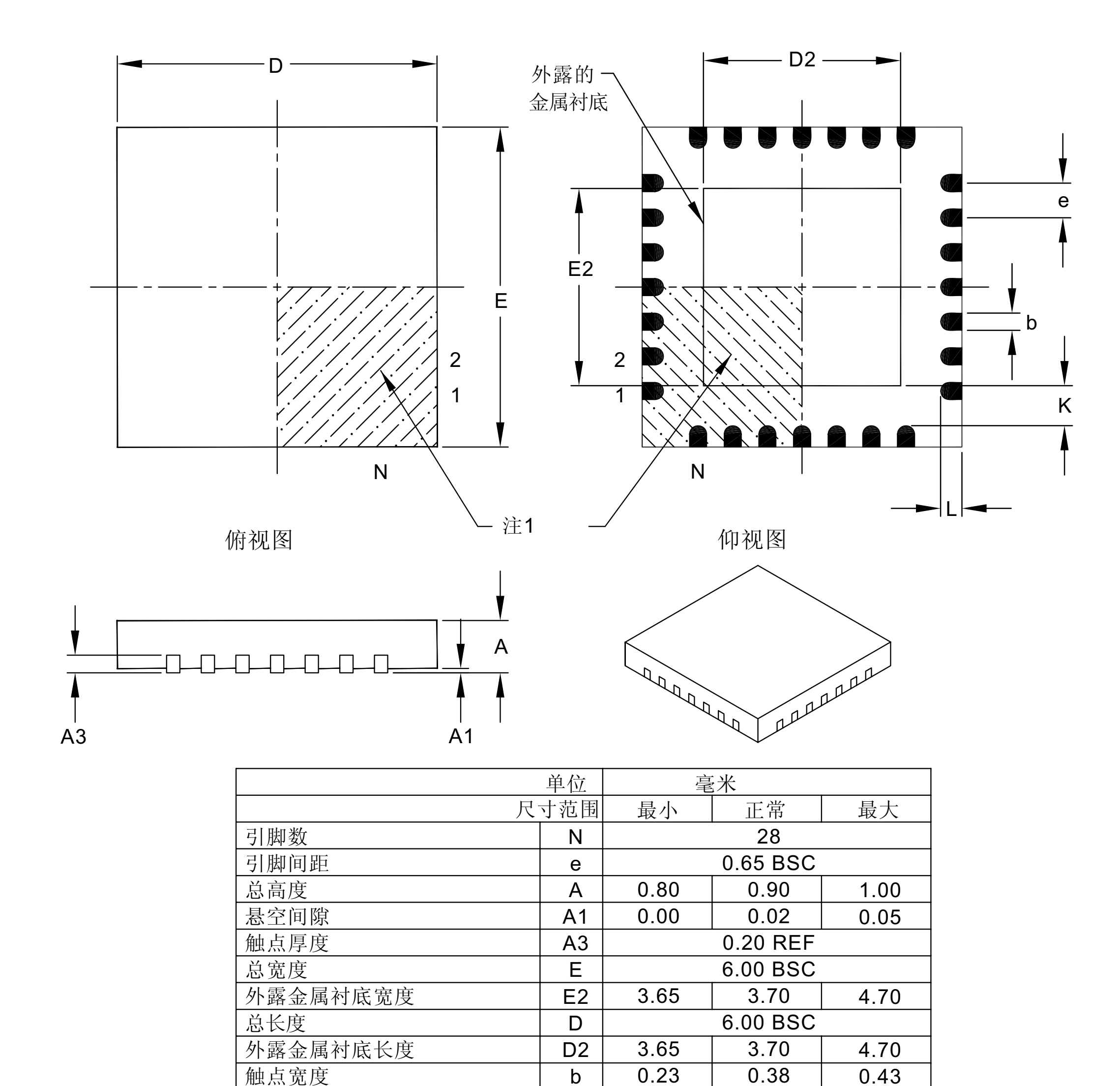
dsPIC30F2010-30I/SO
0648017 e3

图注：	XX...X	客户信息
	Y	年份代码（日历年的最后一位数字）
	YY	年份代码（日历年的最后两位数字）
	WW	星期代码（一月一日的星期代码为“01”）
	NNN	以字母数字排序的追踪代码
	e3	雾锡（Matte Tin， Sn）的 JEDEC 无铅标志
	*	表示无铅封装。 JEDEC 无铅标志（e3）标示于此种封装的外包装上。

注： Microchip 元器件编号如果无法在同一行内完整标注，将换行标出，因此会限制表示客户信息的字符数。

28 引脚塑封四方扁平无脚封装 （MM）——6x6x0.9 mm 主体 （QFN-S），触点长度为 0.40 mm

注：	最新封装图请至 http://www.microchip.com/packaging 查看 Microchip 封装规范。



单位		毫米		
尺寸范围		最小	正常	最大
引脚数	N		28	
引脚间距	e		0.65 BSC	
总高度	A	0.80	0.90	1.00
悬空间隙	A1	0.00	0.02	0.05
触点厚度	A3		0.20 REF	
总宽度	E		6.00 BSC	
外露金属衬底宽度	E2	3.65	3.70	4.70
总长度	D		6.00 BSC	
外露金属衬底长度	D2	3.65	3.70	4.70
触点宽度	b	0.23	0.38	0.43
触点长度 §	L	0.30	0.40	0.50
触点到外露金属衬底的距离 §	K	0.20	—	—

注：
1. 引脚1的可视索引特征可能会有变化，但必须位于阴影区域内。
2. § 重要特性
3. 封装为切割分离
4. 尺寸和公差参考ASME Y14.5M

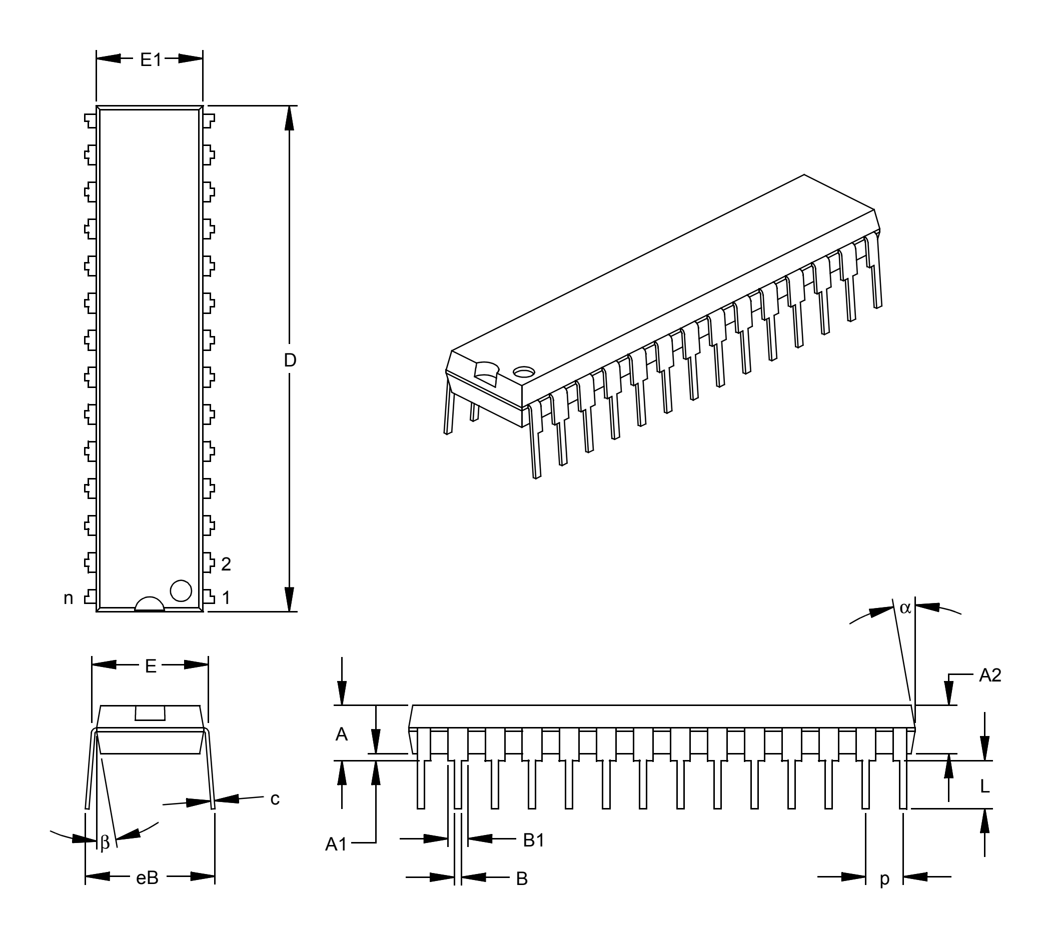
   BSC：基本尺寸。给出的是理论上的精确值，没有公差。

   REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology，图号：C04–124，2006年9月8日

28 引脚窄条塑封双列直插式封装——300 mil 主体 （PDIP）

注：	最新封装图请至 http://www.microchip.com/packaging 查看 Microchip 封装规范。

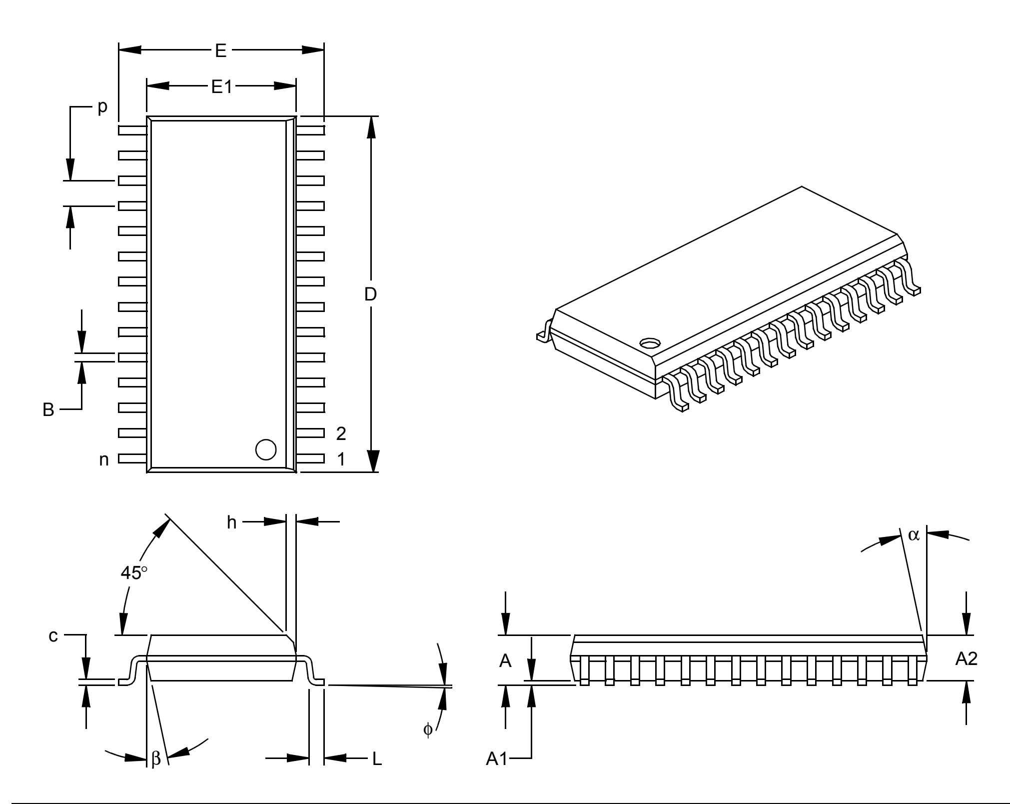


	单位	英寸 *			毫米		
	尺寸范围	最小	正常	最大	最小	正常	最大
引脚数	n		28			28	
引脚间距	p		.100			2.54	
顶端到固定面高度	A	.140	.150	.160	3.56	3.81	4.06
塑模封装厚度	A2	.125	.130	.135	3.18	3.30	3.43
塑模底端到固定面高度	A1	.015			0.38		
肩到肩宽度	E	.300	.310	.325	7.62	7.87	8.26
塑模封装宽度	E1	.275	.285	.295	6.99	7.24	7.49
总长度	D	1.345	1.365	1.385	34.16	34.67	35.18
引脚尖到固定面高度	L	.125	.130	.135	3.18	3.30	3.43
引脚厚度	c	.008	.012	.015	0.20	0.29	0.38
引脚上部宽度	B1	.040	.053	.065	1.02	1.33	1.65
引脚下部宽度	B	.016	.019	.022	0.41	0.48	0.56
总排列间距 §	eB	.320	.350	.430	8.13	8.89	10.92
塑模顶部锥度	α	5	10	15	5	10	15
塑模底部锥度	β	5	10	15	5	10	15

* 控制参数
§ 重要特性
注:
尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸（0.254mm）。
等同于 JEDEC 号: MO-095
图号: C04-070

28 引脚塑封小外形封装——宽条， 300 mil 主体 （SOIC）

注：	最新封装图请至 http://www.microchip.com/packaging 查看 Microchip 封装规范。



单位		英寸 *			毫米		
尺寸范围		最小	正常	最大	最小	正常	最大
引脚数	n		28			28	
引脚间距	p		.050			1.27	
总高度	A	.093	.099	.104	2.36	2.50	2.64
塑模封装厚度	A2	.088	.091	.094	2.24	2.31	2.39
悬空间隙　§	A1	.004	.008	.012	0.10	0.20	0.30
总宽度	E	.394	.407	.420	10.01	10.34	10.67
塑模封装宽度	E1	.288	.295	.299	7.32	7.49	7.59
总长度	D	.695	.704	.712	17.65	17.87	18.08
斜面投影距离	h	.010	.020	.029	0.25	0.50	0.74
底脚长度	L	.016	.033	.050	0.41	0.84	1.27
底脚倾角	φ	0	4	8	0	4	8
引脚厚度	c	.009	.011	.013	0.23	0.28	0.33
引脚宽度	B	.014	.017	.020	0.36	0.42	0.51
塑模顶部锥度	α	0	12	15	0	12	15
塑模底部锥度	β	0	12	15	0	12	15

* 控制参数
§ 重要特性

注:
尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸（0.254mm）。
等同于 JEDEC 号: MS-013
图号: C04-052

附录 A： 版本历史

版本 F （2006 年 5 月）

本数据手册的早期版本中包含超前或初步信息。其中包含不完善的特性数据。

本版本进行了如下更新：

- 支持 I^2C 从地址 （见表 16-1）
- 10 位 A/D 高速转换时序要求（见**第 18.0 节 “10 位高速模数转换器 （ADC）模块”**）
- 工作电流 （I_{DD}）规范 （见表 22-5）
- 空闲电流 （I_{IDLE}）规范 （见表 22-6）
- 掉电电流 （I_{PD}) ）规范 （见表 22-7）
- I/O 引脚输入规范 （见表 22-8）
- BOR 电压限制 （见表 22-10）
- PLL 时钟时序规范 （见表 22-14）
- PLL 抖动规范 （见表 22-15）
- 内部 RC 精度规范 （见表 22-17）
- 看门狗定时器超时限制 （见表 22-20）
- 整个文档中还包含一些其他较小的修正

版本 G （2006 年 12 月）

此版本包含了对封装图的更新。

注：

索引

数字

10 位高速 A/D
  A/D 采集要求 117
  ADCHS 111
  ADCON1 111
  ADCON2 111
  ADCON3 111
  ADCSSL 111
  ADPCFG 111
  CPU 空闲模式下的操作 118
  CPU 休眠模式下的操作 118
  采集要求 117
  掉电模式 118
  复位的影响 118
  寄存器映射 120
  结果缓冲器 112
  连接注意事项 119
  配置模拟端口引脚 119
  输出格式 118
  选择转换时钟 113
  选择转换序列 112
  中止转换 113
  转换操作 112
  转换触发的编程设定 113
10 位高速模数转换器。*参见* A/D
16 位递增 / 递减位置计数器模式 76
  错误检查 76
  计数方向状态 76

A

A/D 111
  1 Msps 时的配置指南 115
  600 ksps 时的配置指南 116
  750 ksps 时的配置指南 116
  参考电压原理图 115
  转换速度 114
  转换速率参数 114
AC 温度和电压特性 155
AC 特性 155
  负载条件 155
ADC
  配置模拟端口引脚 54

B

BOR 特性 154
BOR。*参见*欠压复位。
备用 16 位定时器 / 计数器 77
备用向量表 41
编程 135
编程操作 45
  擦除程序存储器的一行 45
  启动编程过程 46
  闪存程序存储器算法 45
  装载写锁存器 46
编程模型 10
  原理图 11
变更通知客户服务 197
边沿对齐 PWM 85
表指令操作概要 43

C

C 编译器
  MPLAB C18 144
  MPLAB C30 144
CLKO 和 I/O 时序
  特性 159
  要求 159
CPU 空闲模式期间的输出比较工作 73
CPU 空闲模式下的 PWM 操作 89
CPU 休眠模式期间的输出比较工作 73
CPU 休眠模式下的 PWM 操作 89
操作码描述中使用的符号 136
程序存储器和 EEPROM 特性 154
程序地址空间 19
  表指令
    TBLRDL 21
    TBLWTL 21
  存储器映射 19
  构成 20
  使用表指令，访问程序存储器中的数据 21
  数据访问，地址生成 20
程序计数器 10
程序空间
  表指令
    TBLRDH 21
    TBLWTH 21
  使用程序空间可视性，来自程序存储器映射的数据访问 22
  数据空间通过可视性页映射到程序空间的操作 23
程序空间可视性
  程序空间窗口可视操作 23
程序数据表访问 22
除法支持 12
串行外设接口。*参见* SPI
从休眠和空闲中唤醒 41
从休眠中唤醒 121
存储器组织 19

D

DC 特性 147
  BOR 154
  程序存储器和 EEPROM 154
  掉电电流 （IPD） 151
  工作电流 （IDD) 149
  I/O 引脚输出规范 153
  I/O 引脚输入规范 152
  空闲电流 （IIDLE） 150
  欠压复位 153
  温度和电压特性 147
DSP 引擎 13
  乘法器 15
代码保护 121
代码实例
  数据 EEPROM 读 49
  数据 EEPROM 块擦除 50
  数据 EEPROM 块写操作 52
  数据 EEPROM 字擦除 50
  数据 EEPROM 字写操作 51
代码示例
  擦除程序存储器的一行 45
  启动编程过程 46
  装载写锁存器 46

带隙启动时间
时序特性 162
要求 162
单脉冲 PWM 操作 87
单一 PWM 模式
周期 73
低功耗模式
空闲 132
地址发生器单元 31
电机控制 PWM 模块 81
故障时序特性 168
时序特性 168
时序要求 168
电气特性
AC 155
DC 147
掉电电流 （IPD） 151
独立 PWM 输出 87
端口读 / 写操作示例 54

F

防止 OSCCON 意外写入 126
防止意外写入 OSCCON 126
封装信息 185
标识 185
复位 121, 127
具有长晶振起振时间的 POR 129
POR， FSCM 和 PWRT 禁止时的工作 129
欠压复位 （BOR），可编程 129
上电复位 （POR） 127
复位时序特性 160, 161
复位时序要求 161
复位顺序 39
复位源 39
复位源
非法指令陷阱 39
看门狗定时器超时 39
欠压复位 （BOR） 39
未初始化的 W 寄存器陷阱 39
陷阱锁定 39
负载条件 155

G

功能框图
dsPIC30F2010 6
DSP 引擎 14
公式
A/D 转换时钟 113
波特率 107
串行时钟速率 100
PWM 分辨率 84
PWM 周期 84
PWM 周期 （递增 / 递减计数模式） 84
工作电流 （IDD） 149
工作频率与电压
dsPIC30FXXXX-20 （扩展） 147
固件指令 135

H

互补 PWM 操作 86
汇编器
MPASM 汇编器 144

I

I/O 端口
并行 （PIO） 53
I/O 引脚规范
输出 153
输入 152
I^2C 95
I^2C 10 位从模式的工作 97
I^2C 10 位从器件模式的工作
发送 98
接收 98
I^2C 7 位从模式的工作 97
I^2C 7 位从器件模式的工作
发送 97
接收 97
I^2C 模块
编程模型 95
地址 97
寄存器映射 102
总线启动 / 停止位时序特性
从模式 178
总线数据时序要求
从模式 179
总线总线数据时序特性
主模式 176
I^2C 主模式支持 99
I^2C 模块
CPU 休眠和空闲模式期间的工作 101
各种模式 95
功能说明 95
广播呼叫地址支持 99
寄存器 95
IPMI 支持 99
软件控制的时钟延长 （STREN = 1） 99
斜率控制 99
引脚配置 95
中断 99
I^2C 主模式的工作 99
I^2C 主器件模式的工作
波特率发生器 100
多主器件通讯、总线冲突与总线仲裁 100
发送 100
接收 100
时钟仲裁 100

J

寄存器配置
寄存器映射 133
简单捕捉事件模式 68
缓冲器操作 68
霍尔传感器模式 68
Timer2 和 Timer3 选择模式 68
预分频器 68
简单 OC/PWM 模式时序要求 167
简单 PWM 模式 72
输入引脚故障保护 72
简单输出比较匹配模式 72
禁止 UART 105

K

开发支持 143
看门狗定时器
时序特性 160
时序要求 161

看门狗定时器 （WDT） 121, 131
操作 131
使能和禁止 131
勘误表 4
可编程 121
可编程数字噪声滤波器 77
客户通知服务 197
客户支持 197
空闲电流 （IIDLE） 150
快速保存现场 41
框图
10 位高速 A/D 功能 111
16 位 Timer1 模块 58
复位系统 127
共用端口结构 53
I2C 96
SPI 92
SPI 主 / 从器件连接 92
UART 发送器 103
UART 接收器 104
外部上电复位电路 129
振荡器系统 123

M

Microchip 因特网网站 197
MPLAB ASM30 汇编器、链接器和库管理器 144
MPLAB ICD 2 在线调试器 145
MPLAB ICE 2000 高性能通用在线仿真器 145
MPLAB PM3 器件编程器 145
MPLAB REAL ICE 在线仿真器系统 145
MPLAB 集成开发环境软件 143
MPLINK 目标链接器 /MPLIB 目标库管理器 144
模块原理图
输入捕获模式 67
模寻址 33
操作示例 34
起始和结束地址 33
W 地址寄存器选择 33
应用 35

N

内部集成电路。*参见* I2C
内部时钟时序示例 158
内核寄存器映射 27
内核架构
概述 9

O

OC/PWM 模块时序特性 167

P

PICSTART 2 开发编程器 146
PICSTART Plus 开发编程器 146
PLL 时钟时序特性 157
PORTB
寄存器映射 55
POR。*参见*上电复位。
产品标识体系 199
PWM
寄存器映射 90
PWM FLTA 引脚 88
故障状态 88
模式 88
锁存 88
逐个周期 88
使能位 88
PWM 特殊事件触发器
后分频器 89
PWM 更新锁定 88
PWM 时基 83
单事件模式 83
后分频器 84
连续递增 / 递减计数模式 83
预分频器 84
双更新模式 84
自由运行模式 83
PWM 输出改写 87
互补输出模式 87
同步 87
PWM 输出和极性控制 88
输入引脚控制 88
PWM 特殊事件触发器 89
PWM 占空比周期比较单元 85
占空比周期缓冲寄存器 86
PWM 周期 84

Q

QEA/QEB 输入特性 169
QEI 模块
CPU 休眠模式下的定时器操作 77
CPU 空闲模式下的操作 78
CPU 空闲模式下的定时器操作 78
CPU 休眠模式下的操作 77
寄存器映射 79
索引脉冲时序特性 170
索引脉冲时序要求 170
外部时钟时序要求 165
器件概述 5
器件 ID 存储单元 121
器件配置寄存器 132
FBORPOR 132
FGS 132
FOSC 132
FWDT 132
器件指令 135
欠压复位 121
时序要求 161
特性 153

R

RCON 寄存器的初始状态，情形 1 130
RCON 寄存器的初始状态，情形 2 130
RTSP 操作 44
软件堆栈指针，帧指针 10
CALL 堆栈帧 27
软件模拟器 （MPLAB SIM） 144

S

SPI 91
SPI 模块
功能说明 91
SPI 模块 91
CPU 空闲模式期间的工作 93
CPU 休眠模式期间的工作 93
从动选择同步 93
SDOx 禁止 91
帧 SPI 支持 91
字和字节通讯 91
SPI 模式
SPI1 寄存器映射 94

闪存程序存储器 43
控制寄存器 44
NVMADR 44
NVMADRU 44
NVMCON 44
NVMKEY 44
上电定时器
时序特性 160
时序要求 161
上电复位 （(POR）
上电延时定时器 （PWRT） 121
上电复位 （POR） 121
振荡器起振定时器 （OST） 121
省电模式 131
休眠 131
省电模式 （休眠和空闲） 121
使能和设置 UART
设定数据、奇偶校验和停止位选择 105
使能 UART 105
时序特性
A/D 转换
10 位高速 （CHPS = 01、 SIMSAM = 0、
ASAM = 0、 SSRC = 000） 182
10 位高速 （CHPS = 01、 SIMSAM = 0、
ASAM = 1、 SSRC = 111、 SAMC = 00001） 183
A、 B 和 C 类型定时器外部时钟 163
CLKO 和 I/O 159
带隙启动时间 162
电机控制 PWM 模块 168
电机控制 PWM 模块故障 168
复位 160
I2C 总线启动 / 停止位
主模式 176
I2C 总线数据
从模式 178
看门狗定时器 160
OC/PWM 模块 167
PLL 时钟 157
QEI 模块索引脉冲 170
SPI 模块
从模式 （CKE = 0） 173
从模式 （CKE = 1） 174
主模式 （CKE = 0） 171
主模式 （CKE = 1） 172
上电定时器 160
输出比较模块 166
输入捕捉 （CAPX） 166
TimerQ （QEI 模块）外部时钟 165
外部时钟 155
振荡器起振定时器 160
时序图
上电延时时序 （MCLR 连接至 VDD） 128
上电延时时序 （MCLR 未接至 VDD），情形 1 128
上电延时时序 （MCLR 未接至 VDD），情形 2 128
死区时间 87
时序图表和特性
DC 特性 - 内部 RC 精度 158
时序图表。参见时序特性
时序要求
A/D 转换
高速 184
A 类型定时器外部时钟 163
B 类型定时器外部时钟 164
CLKO 和 I/O 159
C 类型定时器外部时钟 164
带隙启动定时器 162
电机控制 PWM 模块 168
复位 161
I2C 总线数据 （主模式） 177
简单 OC/PWM 模式 167
看门狗定时器 161
QEI 模块
索引脉冲 170
外部时钟 165
欠压复位 161
SPI 模块
从模式 （CKE = 0） 173
SPI 模块
从模式 （CKE = 1） 175
主模式 （CKE = 0） 171
主模式 （CKE = 1） 172
上电定时器 161
输出比较模块 166
输入捕捉 166
外部时钟 156
振荡器起振定时器 161
正交解码器 169
时序原理图
边沿对齐 PWM 85
PWM 输出 73
中心对齐 PWM 85
输出比较模块 71
时序特性 166
时序要求 166
输出比较模式
寄存器映射 74
输出比较中断 73
数据地址空间
存储器映射 23, 24
对齐 26
空间 26
宽度 26
MCU 和 DSP （MAC 类）指令 25
Near 数据空间 27
软件堆栈 27
数据 EEPROM 存储器 49
擦除 50
擦除，块 50
擦除，字 50
读 49
防止错误写入 52
写 51
写校验 52
写，块 52
写，字 51
数据累加器和加法器 / 减法器 15
回写 16
舍入逻辑 16
数据空间写饱和 17
溢出和饱和 15
输入捕获中断
寄存器映射 70
输入捕捉模块 67
中断 69
输入捕捉时序要求 166
输入捕捉 （CAPX）时序特性 166
输入电平变化模块
寄存器映射 （bit 15-8） 55
输入电平变化通知模块 54

输入特性
QEA/QEB 169
双输出比较匹配模式 72
单脉冲模式 72
连续脉冲模式 72
死区时间发生器 86
范围 86

T

TimerQ （QEI 模块）外部时钟时序特性 165
Timer2/3 模块 61
16 位定时器模式 61
32 位定时器模式 61
32 位同步计数器模式 61
ADC 事件触发器 64
定时器预分频器 64
寄存器映射 65
门控操作 64
休眠模式下的操作 64
中断 64
Timer2 和 Timer3 选择模式 72
Timer1 模块 57
16 位定时器模式 57
16 位同步计数器模式 57
16 位异步计数器模式 57
寄存器映射 60
门控操作 58
实时时钟 59
振荡器操作 59
中断 59
休眠模式下的操作 58
预分频器 58
中断 59
桶形移位器 17
通用异步收发器。*参见* UART。

U

UART
使能和设置 UART
备用 I/O 105
UART1 寄存器映射 109
自动波特率支持 107
UART 工作
空闲模式 108
休眠模式 108
UART 模块
8 位数据模式下的发送 105
9 位数据模式下的发送 105
波特率发生器 107
地址检测模式 107
发送缓冲器 （UxTXB） 105
发送数据 105
发送中断 106
发送中止符 106
概述 103
环回模式 107
接收错误处理 106
接收缓冲器溢出错误 （OERR 位） 106
接收缓冲器 （UxRXB） 106
接收数据 106
接收中断 106
接收中止符 107
奇偶校验错误 （PERR） 107
使能和设置 105
在 8 位或 9 位数据模式下接收 106
在 CPU 休眠和空闲模式下的工作 108
帧错误 （FERR） 107

W

WWW 地址 197
WWW 在线技术支持 4
外部时钟时序特性
类型 A、 B 和 C 定时器 163
外部时钟时序要求 156
A 类型定时器 163
B 类型定时器 164
C 类型定时器 164
外部引脚描述 7
外部中断请求 41
位反转寻址 35
实例 35
实现 35
修改量表 36
序列表 （16 个位） 36
位置测量模式 77
温度和电压特性
AC 155
DC 147

X

系统集成 121
寄存器映射 133
陷阱 39
来源 39
地址错误陷阱 40
堆栈错误陷阱 40
数学错误陷阱 39
振荡器故障陷阱 40
硬与软 40
休眠和空闲模式期间的输入捕捉工作 69
CPU 空闲模式 69
CPU 休眠模式 69

Y

因特网地址 197
原理框图
PWM 82
输出比较模式 71
正交编码器接口 75
运行时自编程 （RTSP） 43

Z

在线串行编程 （ICSP） 43, 121
在线调试器 （ICD 2） 132
振荡器
配置
低功耗 RC （LPRC） 125
快速 RC （FRC） 125
振荡器工作模式表 122
振荡器配置 124
初始时钟源选择 124
故障保护时钟监测器 126
LP O 振荡器控制 125
配置
锁相环 （PLL） 125
起振定时器 （OST） 124
振荡器起振定时器
时序特性 160
时序要求 161
振荡器选择 121
正交编码器接口逻辑 76
正交编码器接口中断 78

正交编码器接口 （QEI）模块 75
正交解码器时序特性.. 169
指令集 135
汇总表 138
指令寻址模式.. 31
MAC 指令 32
MCU 指令 32
其他指令.. 32
文件寄存器指令 31
支持的基本模式 31
传送和累加器指令 32
中断 37
中断控制
寄存器映射 42
中断顺序.. 41
中断序列
中断堆栈帧 41
中断优先级 38
中心对齐 PWM 85
状态寄存器 10
自动时钟延长... 98
10 位寻址时 （STREN = 1）.................................... 98
7 位寻址时 （STREN = 1）...................................... 98
发送模式.. 98
接收模式.. 98

MICROCHIP 网站

Microchip 网站 （www.microchip.com）为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息：

- **产品支持**——数据手册和勘误表、应用笔记和样本程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题 （FAQ）、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip 的变更通知客户服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请登录 Microchip 网站 www.microchip.com，点击 “变更通知客户 （Customer Change Notification）”服务后按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 （FAE）
- 技术支持

客户应联系其代理商、代表或应用工程师 （FAE）寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过http://support.microchip.com获得网上技术支持。

读者反馈表

我们努力为您提供最佳文档，以确保您能够成功使用 Microchip 产品。如果您对文档的组织、条理性、主题及其他有助于提高文档质量的方面有任何意见或建议，请填写本反馈表并传真给我公司 TRC 经理，传真号码为 86-21-5407-5066。

请填写以下信息，并从下面各方面提出您对本文档的意见。

致: TRC 经理 总页数 ________

关于: 读者反馈

发自: 姓名 ________________

公司 ________________

地址 ________________

国家 / 省份 / 城市 / 邮编 ________________

电话: (______) ________________ 传真:(______) ________________

应用(选填):

您希望收到回复吗?是____ 否____

器件: dsPIC30F2010 文献编号: DS70118G_CN

问题

1. 本文档中哪些部分最有特色?

____

____

2. 本文档是否满足了您的软硬件开发要求?如何满足的?

____

____

3. 您认为本文档的组织结构便于理解吗?如果不便于理解，那么问题何在?

____

____

4. 您认为本文档应该添加哪些内容以改善其结构和主题?

____

____

5. 您认为本文档中可以删减哪些内容，而又不会影响整体使用效果?

____

____

6. 本文档中是否存在错误或误导信息?如果存在，请指出是什么信息及其具体页数。

____

____

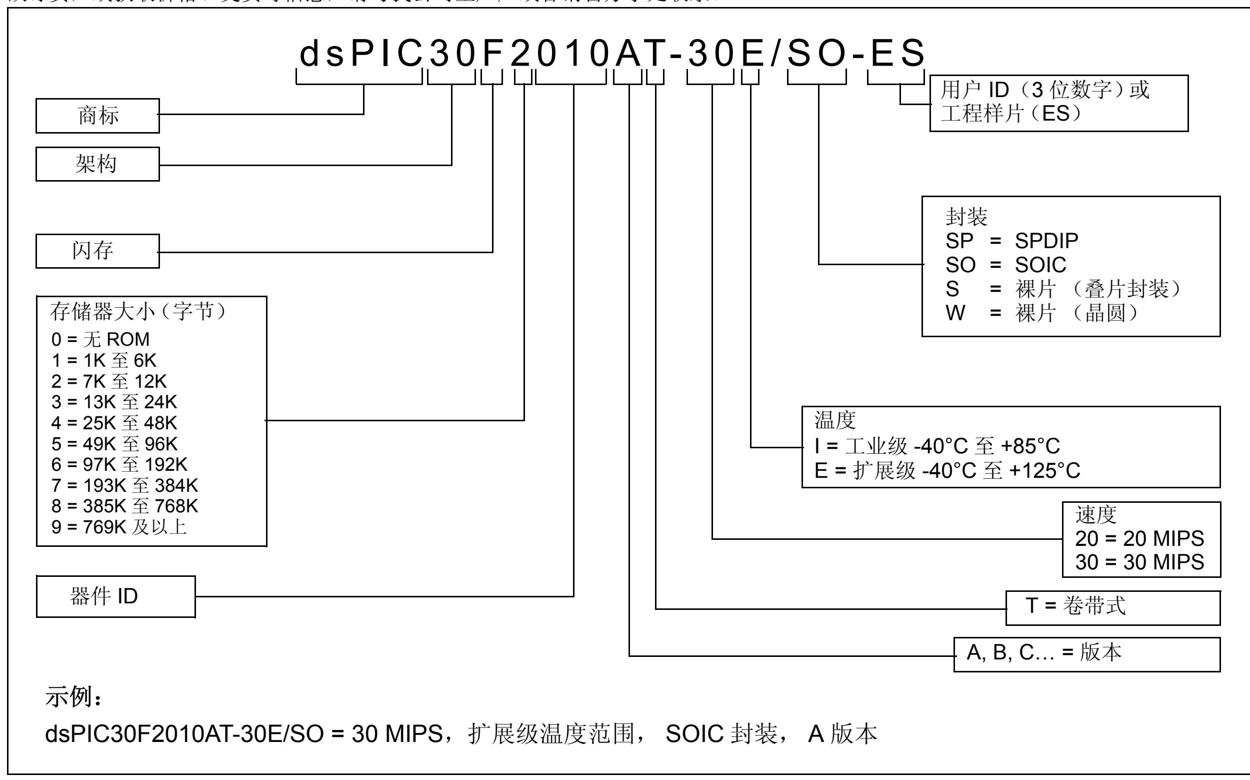
7. 您认为本文档还有哪些方面有待改进?

____

____

产品标识体系

欲订货，或获取价格、交货等信息，请与我公司生产厂或各销售办事处联系。



示例：

dsPIC30F2010AT-30E/SO = 30 MIPS，扩展级温度范围， SOIC 封装， A 版本



全球销售及服务网点

美洲

公司总部 Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277
技术支持：
http://support.microchip.com
网址：www.microchip.com

亚特兰大 Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo
Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

加拿大多伦多 Toronto
Mississauga, Ontario,
Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 – 北京
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 – 成都
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 – 福州
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

中国 – 香港特别行政区
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 – 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 – 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 – 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 – 深圳
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 – 顺德
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

中国 – 武汉
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 – 西安
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

台湾地区 – 高雄
Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 – 台北
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 – 新竹
Tel: 886-3-572-9526
Fax: 886-3-572-6459

亚太地区

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

印度 India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

韩国 Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Penang
Tel: 60-4-646-8870
Fax: 60-4-646-5086

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark-Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

06/25/07